

**Títol:** Planificació, desenvolupament i proves del sistema de gestió logística i administrativa d'una empresa comercial

**Autor:** Marc Arnau i Borràs

**Data:** 27 de juny de 2014

**Director:** Miquel Barceló Garcia

**Departament:** Departament d'Enginyeria de Serveis i Sistemes d'Informació

**Titulació:** Enginyeria Informàtica

**Centre:** Facultat d'Informàtica de Barcelona (FIB)

**Universitat:** Universitat Politècnica de Catalunya (UPC)

SETELSYS	Memòria Projecte Final de Carrera	
----------	-----------------------------------	--

---

## DADES DEL PROJECTE

*Títol del Projecte: **Planificació, desenvolupament i proves del sistema de gestió logística i administrativa d'una empresa comercial***

*Nom de l'estudiant:* Marc Arnau i Borràs

*Titulació:* Enginyeria Informàtica

*Crèdits:* 37,5

*Director:* Miquel Barceló Garcia

*Departament:* Departament d'Enginyeria de Serveis i Sistemes d'Informació

---

## MEMBRES DEL TRIBUNAL *(nom i signatura)*

*President:* Xavier Burgués Illa

*Vocal:* Esteve Codina Sancho

*Secretari:* Miquel Barceló Garcia

---

## QUALIFICACIÓ

*Qualificació numèrica:*

*Qualificació descriptiva:*

*Data:*

---

## Taula de continguts

<b>1.</b>	<b>Introducció.....</b>	<b>8</b>
1.1	Motivació.....	8
1.2	Objectius.....	9
1.3	Organització de la memòria.....	11
<b>2.</b>	<b>Punt de partida .....</b>	<b>12</b>
2.1	L'empresa .....	12
2.2	Sistemes informàtics presents.....	13
2.2.1	Programari ERP principal.....	13
2.2.2	Subsistema d'aplicacions en Access.....	14
2.3	Problemàtica i necessitats derivades de la situació actual .....	16
2.3.1	Deficiències detectades a nivell arquitectònic o estructural .....	17
2.3.2	Deficiències detectades a nivell funcional .....	19
<b>3.</b>	<b>Decisions tècniques .....</b>	<b>20</b>
3.1	Selecció del software de desenvolupament i testing.....	20
3.1.1	Software de desenvolupament .....	20
3.1.2	Sistema gestor de bases de dades (SGBD) escollit .....	24
3.1.3	Software de testing .....	25
3.1.4	Treball en equip i desenvolupament compartit.....	28
3.2	Selecció de la plataforma hardware per allotjar l'aplicació.....	29
3.3	Arquitectura de la solució.....	31
3.4	Salvaguarda de les dades i del servidor .....	35
3.5	Convenis i nomenclatura utilitzada per a la codificació.....	37
3.5.1	Conveni global.....	37
3.5.2	Noms dels controls gràfics de les interfícies d'usuari (UI) .....	38
3.5.3	Nom de les classes.....	38
3.5.4	Nom de les propietats de les classes.....	39
3.5.5	Nom dels mètodes i funcions.....	39
3.5.6	Nom dels delegats, predicats, ... ..	39
3.5.7	Resta d'elements de desenvolupament.....	40
3.5.8	Funcions de test .....	40
<b>4.</b>	<b>Planificació.....</b>	<b>41</b>
4.1	Diagrama de Gantt inicial .....	41
4.2	Descripció de les tasques.....	42
4.2.1	Planificació i Control.....	42
4.2.2	Anàlisi de Requeriments.....	43
4.2.3	Especificació i Disseny .....	44

4.2.4	Desenvolupament .....	45
4.2.5	Testing .....	46
4.2.6	Finalització .....	47
4.2.7	Memòria del Projecte de Final de Carrera .....	47
4.3	<i>Projecte</i> .....	48
<b>5.</b>	<b>Anàlisi de requeriments</b> .....	<b>49</b>
5.1	<i>Inventari d'aplicacions satèl·lit</i> .....	49
5.1.1	Bloc 1: Logística .....	50
5.1.2	Bloc 2: Articles, estocs i inventaris .....	51
5.1.3	Bloc 3: Tarifes i preus .....	52
5.1.4	Bloc 4: CRM, consultes i altres .....	52
5.2	<i>Relacions de dependència entre els diferents blocs d'aplicacions</i> .....	53
5.2.1	Bloc 1: Logística .....	54
5.2.2	Bloc 2: Articles, estocs i inventaris .....	54
5.2.3	Bloc 3: Tarifes i preus .....	55
5.2.4	Bloc 4: CRM, consultes i altres .....	55
5.2.5	Conclusions sobre les dependències entre aplicacions .....	56
5.3	<i>Completesa i vigència de les aplicacions</i> .....	57
5.3.1	Bloc 1: Logística .....	57
5.3.2	Bloc 2: Articles, estocs i inventaris .....	58
5.3.3	Bloc 3: Tarifes i preus .....	58
5.3.4	Bloc 4: CRM, consultes i altres .....	58
5.3.5	Conclusions sobre la completesa i vigència de les aplicacions actuals .....	59
5.4	<i>Funcionalitat actual de les aplicacions vigents</i> .....	60
5.4.1	Aplicació "Pedidos" (Bloc 1: Logística) .....	60
5.4.2	Aplicació "Servir Todo Tiendas" (Bloc 1 – Logística) .....	65
5.4.3	Aplicació "Pendientes" i "Pendientes Y3" (Bloc 1 – Logística) .....	71
5.5	<i>Contrastar les funcionalitats vigents amb el client</i> .....	73
5.6	<i>Ampliació de funcionalitats per part del client</i> .....	74
5.6.1	Aplicació "Pedidos" (Bloc 1 – Logística) .....	74
5.6.2	"Servir Todo Tiendas" (Bloc 1 – Logística) .....	79
5.6.3	Aplicació "Pendientes" i "Pendientes Y3" (Bloc 1 – Logística) .....	83
5.7	<i>Informe previ i revisió de l'anàlisi de requeriments</i> .....	85
<b>6.</b>	<b>Especificació i disseny</b> .....	<b>86</b>
6.1	<i>Models de dades existents en el sistema</i> .....	86
6.1.1	Model de dades de l'ERP principal .....	86
6.1.2	Model de dades de la web comercial .....	93
6.2	<i>Especificació de l'arquitectura del sistema SetelSys</i> .....	98
6.2.1	Diagrama de components i dependències del sistema .....	98
6.2.2	Diagrama de classes del nucli del sistema (SetelSysCore) .....	101

6.3	<i>Especificació del gestor del sistema (SetelSysDataCom).....</i>	103
6.3.1	Casos d'ús.....	103
6.3.2	Diagrama associatiu de classes .....	108
6.4	<i>Disseny de la base de dades de SetelSys .....</i>	109
<b>7.</b>	<b>Desenvolupament .....</b>	<b>111</b>
7.1	<i>Metodologia utilitzada per al desenvolupament .....</i>	111
7.1.1	Eines utilitzades.....	111
7.1.2	Estructuració física dels arxius de l'aplicació .....	112
7.1.3	Automatització de l'entorn de desenvolupament i pre-producció.....	115
7.1.4	Ordre i cronologia en el desenvolupament.....	116
7.1.5	Modificació dels noms dels aplicatius.....	116
7.1.6	Documentació de la codificació .....	117
7.1.7	Premisses i consideracions presents durant el desenvolupament .....	118
7.2	<i>Estructura de la solució .NET.....</i>	119
7.3	<i>Cicle de vida en la implementació de funcionalitats .....</i>	120
7.4	<i>Consideracions preses en el disseny de les interfícies i interaccions amb l'usuari. ...</i>	121
7.4.1	Notificacions amb finestra emergent.....	121
7.4.2	Mecanismes d'ajuda a l'usuari en formularis o pàgines .....	123
7.4.3	Utilització sistemàtica de controls .....	124
7.4.4	Utilització sistemàtica d'imatges identificadores.....	125
7.5	<i>Implementació de la base de dades principal de SetelSys .....</i>	126
7.5.1	Taules per a la gestió del sistema.....	127
7.5.2	Taules per a l'ús d'aplicacions logístiques.....	128
7.5.3	Taules per l'ús d'aplicacions relacionades amb els articles .....	134
7.5.4	Taules per l'ús d'aplicacions relacionades amb les tarifes i preus de venda .....	137
7.6	<i>Implementació dels components del projecte.....</i>	138
7.6.1	SetelSysCore .....	138
7.6.2	SetelSysRunner.....	156
7.6.3	SetelSysWinService .....	170
7.6.4	<i>Maiden: Servei web .....</i>	172
7.6.5	<i>SetelSys: Aplicació web amb Silverlight.....</i>	177
7.7	<i>Implementació d'eines d'ajuda .....</i>	191
7.7.5	Automatització de la instal·lació del servei de Windows.....	198
7.7.6	Automatització de còpies de seguretat de tot el projecte.....	198
7.8	<i>Llibreries i components externs utilitzats .....</i>	199
7.8.1	Components .NET FrameWork.....	199
7.8.2	Llibreries externes .....	200
<b>8.</b>	<b>Testing .....</b>	<b>201</b>
8.1	<i>Metodologia: Test Management Approach (TMap) .....</i>	201

8.1.1	Què és TMap? .....	202
8.1.2	Cicle de vida segons TMap .....	202
8.2	<i>Aplicació de TMap en el Projecte</i> .....	203
8.2.1	Fase de planificació .....	203
8.2.2	Fase de preparació .....	203
8.2.3	Fase d'especificació .....	204
8.2.4	Fase d'execució .....	204
8.2.4.1	Gestió d'errors.....	204
8.3	<i>Fase de conclusió</i> .....	205
8.3.1	Fase de Infraestructura .....	205
8.3.2	Control.....	205
<b>9.</b>	<b>Automatització</b> .....	<b>206</b>
9.1	<i>Estratègia utilitzada</i> .....	206
9.2	<i>Eines utilitzades</i> .....	206
9.3	<i>Patró de disseny de proves: Page Objects Pattern</i> .....	206
<b>10.</b>	<b>Anàlisi econòmic</b> .....	<b>207</b>
<b>11.</b>	<b>Conclusions</b> .....	<b>209</b>
11.1	<i>Possibles millores futures</i> .....	211
<b>12.</b>	<b>Bibliografia i referències</b> .....	<b>212</b>

# 1. Introducció

## 1.1 Motivació

**L'objectiu principal d'aquest projecte és el de donar solució a una necessitat real en una empresa determinada, per a poder millorar i optimitzar els seus processos informatitzats.**

Dit així, pot semblar un resum fred i poc passional, però la veritat és que aquest projecte ens brinda la oportunitat d'analitzar la solució a un escenari prou complex, utilitzant eines i recursos coneguts, així com altres de nous necessaris per aconseguir donar a llum un projecte que requereix una àmplia gama de coneixements i característiques, amb la dificultat afegida de que no es parteix des de zero, sinó que l'empresa en qüestió ja disposa d'un sistema actualment en producció.

Com a motivació del projecte ens proposem un seguit de punts a potenciar i treballar:

- Aplicar les **tècniques** i l'**arquitectura** necessària per donar una solució modificable, adaptable i multi plataforma, acotant els costos de desenvolupament i proves, així com el de desplegament una vegada el sistema està en producció.
- Establir un nivell de **feedback** adequat amb el client final, per a focalitzar i determinar al màxim les necessitats reals desitjades al sistema.
- Assegurar una òptima **qualitat** en la solució proposada, garantint un alt nivell de cobertura de proves, i a la vegada, plantejant un conjunt de proves reutilitzables per a validar futures extensions del sistema, així com correccions que s'hagin de realitzar.
- **Planificar** l'anàlisi i desenvolupament del projecte tant acuradament com ens sigui possible, per tal d'obtenir un *timeline* el màxim real possible i minimitzar les desviacions, tant en temps com en recursos.
- Treballar amb un entorn **dependent** i **satèl·lit**. No partirem d'un escenari en blanc, on tota novetat gira entorn del nostre sistema, sinó que hem d'adaptar el nostre sistema a un altre de principal, i per tant controlar i planificar els canvis que es produeixen en ell per evitar problemes col·laterals en el nostre.
- Estudiar, analitzar i escollir les **eines** a utilitzar per construir les nostres idees i el nostre objectiu. Tant a nivell del desplegament, desenvolupament, verificació i testing, etc.

En l'apartat de conclusions podrem analitzar en profunditat el resultat de tot aquest plantejament, però seria sincer afirmar que tot projecte enriqueix i millora l'experiència, les eines i les metodologies aplicades en els anteriors. I aquest no n'és una excepció. Des dels primers anàlisis de requisits fins a la implantació final, no hem deixat d'aprendre i construir una base més ferma i experimentada pel futur, o si més no, això esperem.



## 1.2 Objectius

**SetelSys** pretén convertir-se en un motor de gestió de tots els procediments i aplicacions que actualment s'utilitzen en l'empresa per a donar resposta a les funcionalitats que no estan integrades en el seu sistema principal de gestió, al qual, d'ara en endavant ens referirem a ell com a ERP.

Tot i que actualment existeixen un seguit de programes que donen cabuda a aquestes necessitats (tal i com s'explica en l'apartat "[Punt de Partida](#)" d'aquesta memòria), aquests ho fan de manera precària, descontrolada, poc intuïtiva i eficient.

D'altra banda, SetelSys també haurà d'implementar la capacitat de gestionar processos automatitzats que s'executin de forma diària i parametritzada (tant pel que fa al calendari d'execució com el comportament), i de tot el subsistema necessari per a la gestió de la configuració de l'entorn.

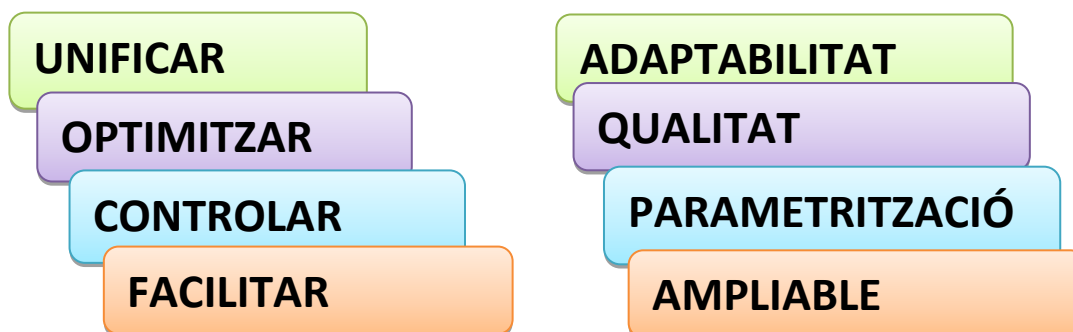
Podríem citar els següents objectius, clarament definits:

- Homogeneïtzar les diferents aplicacions i processos sota una mateixa aplicació. Volem un sistema capaç d'englobar i gestionar els diferents sistemes sota un mateix entorn de treball, i volem eliminar l'existència de solucions independents i disseminades en diferents equips servidors. Tot ha d'estar englobat en una sola solució i un únic servidor de dades i d'aplicacions.
- Permetre un creixement de les funcionalitats existents, així com facilitar la incorporació de noves funcionalitats, agilitzant-ne al màxim el procés, i a la vegada, fer-lo transparent a l'usuari. La incorporació de una nova funcionalitat o d'alguna modificació ha de suposar un cost mínim per a l'usuari.
- Coordinar i controlar les modificacions en les bases de dades de l'ERP, a través de les eines necessàries per a detectar possibles canvis en les bases de dades d'aquest que puguin involucrar problemes en el funcionament i la coherència de les funcionalitats del SetelSys.
- Dotar a l'aplicació d'un sistema de control i registre d'errors, per a poder monitoritzar l'estat del sistema en tot moment, així com poder generar resums diaris automàtics dels resultats de processos automatitzats que actualment no generen cap informació sobre el resultat de la seva execució.
- Control d'accés a les diferents funcionalitats de SetelSys a través d'un sistema d'usuaris propi que permeti definir les polítiques d'accés de cada usuari a cada funcionalitat, i inclús dependent de l'origen de la seva connexió.

- Agilitzar i facilitar la instal·lació i implantació de SetelSys en nous equips. La implantació i l'actualització de SetelSys ha de tenir un cost mínim en temps, complexitat i coneixements de l'usuari final.
- Millorar la qualitat dels processos i les funcionalitats, i això engloba elements com:
  - Assegurar la qualitat global del producte, amb l'objectiu de detectar els errors així que sigui possible, per tal de disminuir-ne el seu cost de correcció.
  - Validar que s'ha implementat una solució correcta en termes d'usabilitat, creant interfícies gràfiques intuïtives, manejables i clares.
  - Assegurar la solidesa de la solució, millorant la robustesa del sistema davant qualsevol tipus de situació imprevista, o ús incorrecte.
  - Implementar una solució amb rendiment correcte, que en permeti un ús fluid i ràpid.
  - Plantejar les proves de forma que es puguin reutilitzar al màxim a posteriori en noves versions de la solució, i a la vegada incorporant l'automatització de proves.
  - I en conjunt, realitzar un enfocament en metodològic i efectiu en termes de testing i qualitat
- Permetre administrar i parametritzar el comportament de les diferents funcionalitats de SetelSys perquè un usuari administrador pugui gestionar el comportament de les diferents funcionalitats sense dependre del desenvolupador. Es tracta d'obrir al màxim la personalització en aquells punts on detectem que és necessari.
- Reduir la dependència en equips de hardware que actualment estan obsolets i que de forma distribuïda allotgen programaris inconnexos i sovint de forma redundant i caòtica. Com més elements de hardware d'aquest tipus eliminem i aconseguim centralitzar la solució en un sol servidor, menys probabilitats hi haurà a l'empresa de paralització del treball per la caiguda d'un dispositiu de hardware.

Com podeu veure, no ens quedem curts en els objectius, però la veritat és que la solució desitjada i les necessitats del client així ho exigeixen. De fet, part d'aquest objectius sorgeixen directament de les necessitats detectades i transmeses per l'empresa, mentre que d'altres apareixen com a objectius derivats de les necessitats o prerequisits per a produir una solució completa i de qualitat, sota uns ulls crítics en l'anàlisi arquitectònic del sistema actual.

Resumint, les 8 paraules claus que sintetitzen els nostres objectius:



### 1.3 Organització de la memòria

El projecte de final de carrera “Setelsys” s’ha dut a terme de forma conjunta amb el meu company Jordi Soldevila i Corominas. Vam optar per a realitzar el projecte conjuntament amb la idea de que fos un projecte més “real”, amb un component de treball en equip i distribuït, amb un afegit interès a nivell de planificació i coordinació.

S’ha optat per un treball conjunt en algunes àrees, però en altres hi ha hagut una clara separació de responsabilitats i tasques. I aquesta, s’ha fet precisament per a aprofitar millor els coneixements de cadascú dins el seu àmbit laboral. No obstant, aquesta separació no ha sigut total, en el sentit d’una caixa negra, sinó que s’ha fet amb un degoteig constant d’informació i coordinació, per a conèixer clarament en què consistia la feina de cadascú i veure com anava evolucionant el projecte.

A grans trets, a nivell conjunt s’han dut a terme les següents àrees:

- Planificació, decisions tècniques i coordinació
- Anàlisi de requeriments, dissenys funcionals
- Conclusions

I a nivell més individual, la fase de Desenvolupament s’ha dut a terme per part del Marc Arnau, i la fase de Testing per part del Jordi Soldevila.

En línia amb l’organització que s’ha mantingut durant el projecte, la memòria d’ambdós projectes s’ha realitzat de forma similiar, per a simplificar i facilitar la lectura i comprensió de com s’ha dut a terme el projecte, així com de la focalització que ha tingut cadascú en la seva part de la feina.

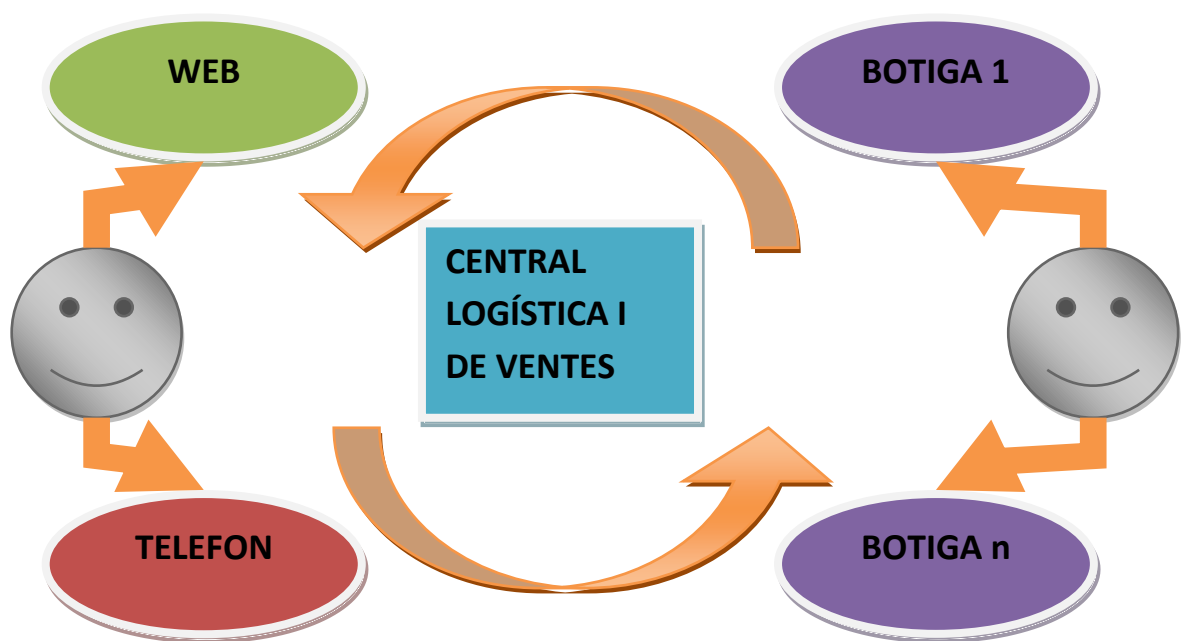
Les dues memòries tenen una part inicial i una part final similars, ja que tot el que és la planificació, anàlisi de requeriments i conclusions, entre altres, s’ha realitzat de forma conjunta. I el que són les àrees específiques del projecte, el seu responsable les ha explicat en detall en la seva memòria, i en la del company s’hi ha afegit un resum, per a mantenir una consistència i coherència, i permetre al lector de la memòria tenir una visió completa del projecte.

## 2. Punt de partida

Primer de tot i per poder entendre la situació inicial del sistema prèviament a la creació del projecte SetelSys, anem a definir els diferents elements i actors que intervenen en l'escenari:

### 2.1 L'empresa

Es dedica a la comercialització d'articles relacionats amb el món de les motocicletes, tant a través de botigues per la venda directa al client, com a través de la seva pàgina web i telecomandes a través de telèfon, tant a nivell nacional com internacional.



Actualment el **client** pot adquirir les seves compres:

- Directament a la **botiga**, en el cas de que disposi d'estoc.
- Fent la **reserva** a la botiga, a la qual s'enviarà el material des de la central.
- Fent la reserva a través de la pàgina **web**, amb la qual cosa s'enviarà directament al client, tot i que disposa de l'opció de recollir-ho en alguna de les botigues disponibles.
- Fent la reserva per **telèfon**, amb un comportament idèntic al d'una comanda via web.

## 2.2 Sistemes informàtics presents

### 2.2.1 Programari ERP principal

L'empresa disposa d'un sistema d'una tercera empresa, utilitzat per a la gestió de la major part de tasques tant a la central com a les botigues.

- **Gestió:** Documents de compra i venda d'articles, estocs, tarifes, magatzems, ...
- **Control de Presència** per al control i organització dels horaris dels seus treballadors.
- **Comptabilitat** vinculada amb la part de gestió.
- **Informes** i resums analítics tant de gestió com comptabilitat.
- Gestió de terminals de **punt de venda** (TPV)
- ...

Aquest ERP es divideix en dos aplicacions independents per cobrir les funcions esmentades:

- **Front:** Utilitzat com a programari de **Punt de Venta** a les botigues, amb funcionalitats limitades per aquest tipus d'activitat.
- **Manager:** Utilitzat com a programari de **gestió** des de la central, tant per la part de la gestió a nivell de compres i ventes, comptabilitat, gestió analítica, tarifes, traspassos entre magatzems, estocs, ...

Pel que fa a l'arquitectura d'aquests programaris, es caracteritza per:

- Bases de dades implementades en un servidor **Microsoft SQL Server**.
- Aplicacions **clients** desenvolupades per a executar-se sobre Microsoft **Windows**.

L'accés dels usuaris a aquestes dues aplicacions es fa a través de la pròpia **xarxa**, utilitzant:

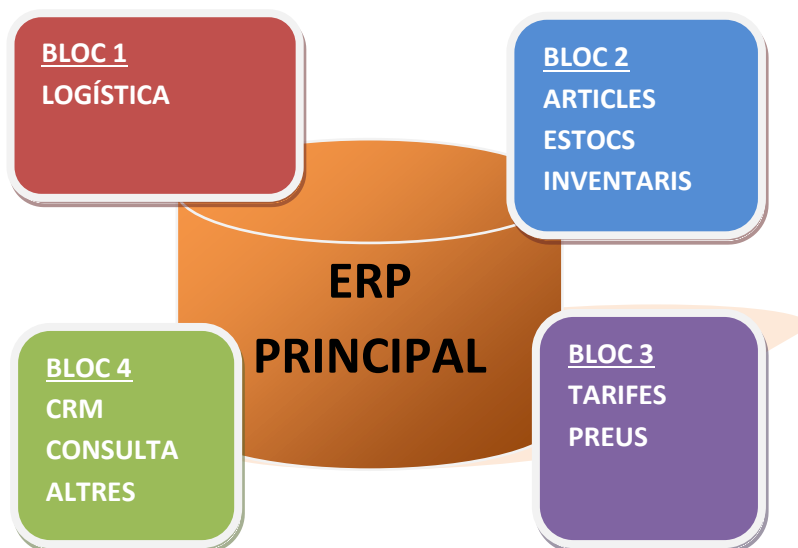
- Clients instal·lats **localment** en els equips dels usuaris.
- Accessos a través del servei de **Terminal Server** cap als servidors de dades on s'executen les aplicacions sobre la sessió en el servidors.

### 2.2.2 Subsistema d'aplicacions en Access

Existeix un subsistema, que s'executa de forma **paral·lela** al ERP principal, conformat per un gran nombre d'arxius de **Microsoft Access**, que treballen amb **vincles** a les bases de dades del ERP i utilitzen taules **auxiliars** de treball, generant arxius excessivament **extensos** i poc eficients, que cobreixen les **mancances** de l'ERP principal, tant en l'aspecte logístic, com el control d'articles, informes, etc.

La lògica ens diria que la forma més fàcil de resoldre aquestes mancances seria que el fabricant del software modifiqués el seu programari o el personalitzés pel client, però com que aquest es nega a procedir per aquesta via per política d'empresa, aquesta es veu obligada a dependre d'un sistema satèl·lit per cobrir les necessitats extres.

Aquests arxius o aplicacions d'Access impliquen un seguit de problemàtiques que analitzarem en el següent punt. De moment però analitzem com s'organitzen a nivell categòric o temàtic aquests aplicatius, en base de l'àrea de l'empresa amb la que es relacionen.



#### *Bloc 1 – Logística*

Compren tots els aplicatius relacionats amb la gestió automatitzada de comandes. Per tal de poder agilitzar el procés d'enviament de comandes als clients (ja sigui per la seva recollida en botiga o bé per l'enviament directe al domicili del client), existeixen un seguit d'aplicacions per a generar els moviments de material entre magatzems o cap al client final, amb la generació dels corresponents albarans, regularització de l'estoc, moviments comptables, etc.

És un dels blocs més complexos, ja que l'algorísmica del procés té en compte molts factors i paràmetres.

SETELSYS	Memòria Projecte Final de Carrera	
----------	-----------------------------------	--

## ***Bloc 2 – Articles, estocs i inventaris***

Compren tots els aplicatius relacionats amb la gestió especialitzada d'articles (creació massiva, gestió d'articles especials, re càlcul de costos, ...), estocs (entrada massiva d'ubicacions d'articles, gestió de l'estoc dels proveïdors, mínims, ...) i també d'inventaris (gestió de regularitzacions).

## ***Bloc 3 – Tarifes i preus***

Compren els aplicatius necessaris per la modificació avançada de tarifes i preus associats als articles i la capacitat de generar còpies automàtiques i sota demanda dels preus. També existeix un aplicatiu per permetre la importació de preus a través d'arxius separats per comes o de Microsoft Excel.

Un dels elements més importants d'aquest bloc és un procés automatitzat encarregat de calcular i regenerar preus de tarifes derivades de tarifes base.

## ***Bloc 4 – CRM, Consulta i Altres***

Engloba aplicatius utilitzats per diferents tasques com la gestió de les targetes de fidelització, així com les consultes enregistrades que fan els clients finals o les pròpies botigues contactant amb la central.

En aquest bloc englobem també aplicacions puntuals i difícils de categoritzar que no tenen cabuda en cap altre bloc.

## 2.3 Problemàtica i necessitats derivades de la situació actual

En primer lloc cal entendre la situació des d'un global.

- Per una part tenim un sistema ERP principal, que dona cabuda a la majoria de necessitats de l'empresa, però està limitat pel que fa al creixement i a la especialització dels processos de l'empresa.
- Per una altra banda disposem d'un subsistema d'aplicacions de Microsoft Access que permet accomplir satisfactòriament els diferents processos interns del seu negoci, però presenta **deficiències** en bastants punts.

L'empresa té clar que el seu motor principal de gestió li funciona adequadament i com necessita, i vol enfocar les millores en la remodelació del sistema satèl·lit d'aplicacions que actualment presenten moltes **deficiències**. Vol construir un sistema satèl·lit robust, fiable i ampliable que cobreixi les necessitats actuals i permeti implementar amb certa "facilitat" necessitats futures.

Degut a la impossibilitat d'adaptar i personalitzar el programari del ERP Principal a les necessitats més concretes de l'empresa, aquesta requereix de la capacitat de disposar d'un sistema informàtic paral·lel i coordinat amb el sistema central actual per tal de poder dur a terme millores en els seus processos de treball que permetin optimitzar-los, millorar-los i unificar-los en una sola plataforma. I aquí, en aquest punt és on entra de ple el nostre projecte SetelSys.



### 2.3.1 Deficiències detectades a nivell arquitectònic o estructural

Existeixen un seguit de deficiències en el sistema satèl·lit actual, pel que fa a l'arquitectura de la distribució com del desenvolupament:

- **Limitació de l'arquitectura Microsoft Access a nivell de base de dades i entorn operatiu:** Sovint ens trobem davant d'aplicacions amb volums de dades suficientment grans com per desestabilitzar el funcionament d'aquest tipus de bases de dades o per ralentitzar-ne el funcionament fins a temps poc acceptables. D'altra banda, els enllaços a taules existents del sistema ERP principal es fan sempre amb connexions ODBC que sovint impliquen dependències en els canvis de les taules originals o de les credencials del domini d'Active Directory.
- **Processos poc eficients i poc segurs a nivell de coherència en cas d'errors:** En els aplicatius actuals no existeix la consideració d'errors, amb la qual cosa quan es produeixen en el procés d'una part del codi, els resultats són fins a cert punt indeterminats i "perillosos". Cal desenvolupar els aplicatius tenint en compte com actuar en cas d'error dins un procés.
- **Manca de control d'errors en temps d'execució:** No hi ha un tractament adequat utilitzant la filosofia *Try/Catch*. Qualsevol error es queda sense tractament. Tampoc es tenen en compte errors d'accessibilitat, de connexió amb la base de dades, violacions de claus primàries o foranies, ... Resumint: El control d'errors no existeix.
- **Distribució ineficient dels aplicatius:** Les diferents aplicacions no estan centralitzades sota un sol servidor, sinó que s'allotgen de manera distribuïda en diferents servidors (3-4 servidors). Això dificulta les tasques d'administració, salvaguarda de les dades, control d'execució, ... La fragmentació de les aplicacions suposa moltes pèrdues de temps a nivell administratiu i implica l'existència d'unes plataformes de hardware innecessàries, obsoletes i innecessàriament redundants.
- **Excés de dependències entre aplicatius:** Molts aplicatius presenten lligams entre ells a través de taules referenciades. Això vol dir que diferents arxius de Microsoft Accés depenen entre ells, creant sovint autèntiques teranyines de dependències. Un canvi en un dels arxius pot generar problemàtiques en molts arxius dependents o lligats a ell.
- **Cost elevat d'implementació de noves funcionalitats:** L'estructura actual d'aplicacions i les dependències anotades en l'anterior punt e) impliquen que la implementació de noves funcionalitats en un aplicatiu sigui molt elevat.
- **Limitació de l'entorn de desenvolupament:** Les eines de la plataforma Microsoft Access limiten les capacitats tecnològiques pel que fa al desenvolupament, tant pel que fa al llenguatge com a la capacitat d'integrar llibreries, elements externs, programació orientada a objectes, tecnologies més actuals i eficients, ...

- **Desconeixement de la vigència de les aplicacions:** Actualment existeixen un seguit d'aplicacions, que els propis usuaris desconeixen si se'n fa un ús o no. Cal determinar quines aplicacions són actuals i funcionals i quines no.
- **Aplicatius repetits per la mateixa funcionalitat:** Un altre element que indueix a la confusió és el fet que existeixin diferents aplicacions que donen la mateixa funcionalitat. Això en gran part és degut a la disseminació i falta de control en el desenvolupament d'aquest sistema satèl·lit.
- **Redundància de dades:** Existeixen aplicatius allotjats de forma repetida, no per una qüestió de balanceig ni de distribució organitzada de dades, sinó per una mala política de gestió d'aquests aplicatius i del sistema en general. Aquesta redundància descontrolada implica que sovint els usuaris treballin amb dades diferents en més d'un arxiu i generi un volum d'informació dividida, que indueix a errors o falta d'informació en molts casos.
- **Dades *off-line*:** Algunes aplicacions no consulten en línia les dades reals del sistema ERP principal, sinó que s'actualitzen un cop al dia a través d'un procés automatitzat. Això fa que certes consultes d'informació no estiguin actualitzades al moment i per tant pot provocar un desfàs informatiu respecte la realitat. Es vol aconseguir que el màxim d'aplicacions siguin amb connexió *on-line* amb les bases de dades principals del ERP.

### *En l'àmbit dels processos automàtics*

Durant les hores de inactivitat del sistema, s'aprofita per executar un seguit de processos automatitzats que poden generar un temps de còmput considerablement alt, i que no es poden executar durant els horaris de treball per tal d'evitar incidències en el rendiment del sistema. Aquests processos tenen aspectes a corregir **molt importants**:

- Optimitzar els temps de còmput, utilitzant consultes més optimitzades i ben estructurades a la B.D.
- Controlar els errors en el procés i enregistrar-los en un diari d'incidències (log) per a consultes posteriors, anotant en tot moment el component, classe i rutina exacta on s'ha produït l'error conjuntament amb el missatge i codi d'error.
- Informar diàriament als administradors dels resultats de les execucions. Caldrà notificar a través de correu electrònic (parametritzant-ne els possibles destinataris), el resultat de l'execució diària dels processos automatitzats, indicant en tot moment si s'han completat correctament i en cas contrari indicant el missatge d'error.
- Dotar d'un sistema que permeti parametritzar quins processos es volen executar de forma automàtica i en quin interval concret (diari, múltiple diari, setmanal, mensual, sota demanda,...). S'haurà de facilitar el procés de configuració dels processos a través d'una interfície de gestió.

- Caldrà elaborar un sistema de màxima disponibilitat. Això vol dir que en cas d'un reinici del servidor, el sistema d'execució de processos automatitzats s'iniciï automàticament a l'iniciar el sistema operatiu on està allotjat i no depengui d'un factor humà per la seva reactivació.

### 2.3.2 Deficiències detectades a nivell funcional

A part de les deficiències a nivell estructural o arquitectònic, existeixen unes mancances actuals a nivell funcional dels aplicatius existents, tant pel que fa a noves necessitats no implementades com la millora de les ja existents. Les podem englobar en els diferents blocs que em subdividit anteriorment:

#### En el "Bloc 1" (Logística)

- Optimitzar el procés d'enviament de les comandes directament a client, generant els corresponents moviments d'estocs entre les botigues i la central per agilitzar-ne el temps d'entrega.
- Millorar el sistema de reserves en les botigues per assegurar un immediat moviment d'estocs per a cobrir les reserves dels clients.
- Evitar moviments d'estocs innecessaris.
- Ajudar en el procés d'organització logística dins del magatzem central. Gestió de caixes i seguiment de caixes de reserva.
- Permetre la proposta de traspassos entre botigues i comandes a proveïdor, basant-se en categories d'articles i les necessitats que porten explícites per cada magatzem per tal d'automatitzar el procés.

#### En el "Bloc 2" (Articles, estocs i inventaris)

- Agilitzar el procés de creació d'articles, amb les seves talles i colors, amb una inicialització ràpida de les tarifes per defecte i estocs mínims i màxims pels diferents magatzems.
- Sistema de gestió de les faltes d'estoc en els diferents magatzems per tal de poder ajudar en els processos de regularització d'estoc i no computar estocs erronis en els moviments de material entre les diferents botigues i la central.
- Automatitzar la captura d'ubicacions dels articles al magatzem central, que actualment requereix un treball molt manual.
- Permetre el recàlcul dels costos dels articles per ajustar el seu cost d'estoc, en base dels diferents documents de compra que existeix en l'històric de la base de dades.

#### En el "Bloc 3" (Tarifes i preus)

- Permetre generar còpies manuals i automàtiques de tarifes de preus de forma que un usuari pugui accedir a la còpia i restauració amb una interfície senzilla i fàcil d'utilitzar.
- Parametritzar el procés de regeneració de preus i tarifes d'articles, de forma que es puguin escollir quines tarifes es volen regenerar en base de les tarifes bàsiques.

#### En el "Bloc 4" (CRM, Consulta i Altres)

- Dotar d'un sistema potent de consulta de comandes pendents de servir al client, amb la màxima informació sobre el seu estat present i futur.

### 3. Decisions tècniques

#### 3.1 Selecció del software de desenvolupament i testing

##### 3.1.1 Software de desenvolupament

Tenint en compte les següents premisses pel que fa als requisits de la solució a desenvolupar i implementar:

- *Solució multi plataforma i portable*: existeixen equips clients amb diferents sistemes operatius.
- *Actualització ràpida i àgil del sistema per a tots els clients*: les actualitzacions s'han de desplegar amb la màxima transparència i la mínima intervenció tècnica i de l'usuari.
- *Ràpid desplegament en nous equips clients*: poder instal·lar amb la màxima facilitat la solució minimitzant els costos administratius.
- *Disposar de les eines necessàries per a crear una aplicació prou complexa com la de SetelSys*.
- *Afinitat en l'ús de les eines*: no hem de sacrificar cap potencial ni possibilitat en l'elecció, però en igualtat de condicions, és important poder desenvolupar l'aplicació amb una eina més coneguda i familiar.

El primer plantejament que cal fer-nos és l'avaluació sobre si la solució ha de ser una aplicació d'escriptori o bé una solució basada en la web. D'aquest plantejament en surt el primer anàlisi d'avantatges i inconvenients:

APLICACIÓ D'ESCRITORI (Client local)	
Avantatges (+)	Inconvenients (-)
➤ Menys càrrega en els servidors	➤ Major requeriment de maquinari en els equips clients.
➤ Desenvolupament més ràpid	➤ Implantació més lenta
➤ Interfície més potent i fàcil de personalitzar.	➤ Dependència de plataforma (sistema operatiu i versió)
	➤ En cas d'haver d'aplicar una nova actualització és necessari instal·lar-la en cada equip de la xarxa o desenvolupar un sistema d'actualitzacions automàtiques.

APLICACIÓ WEB	
Avantatges (+)	Inconvenients (-)
➤ No requereix una instal·lació assistida. Simplement tenir un navegador web actualitzat.	➤ Necessari verificar compatibilitat amb tots els navegadors (o si més no els usats en l'empresa)
➤ Implantació immediata	➤ Requereix d'un SERVIDOR WEB físic, encarregat de donar servei a la INTRANET.
➤ Portabilitat màxima (independència de plataforma i de versió)	
➤ Menys tràfic de dades, solament s'envia la informació final al client, la resta s'executa en el servidor a través de la xarxa local.	
➤ Actualitzacions automàtiques, sense necessitat de distribuir.	
➤ Major coneixement i especialització en la plataforma de desenvolupament.	

En base a les avantatges i inconvenients presentats, ens vàrem decantar per seleccionar com a plataforma de l'aplicació el treballar amb una **APLICACIÓ WEB**, principalment pels següents motius:

- Compleix les tres primeres premisses que hem anotat com a requisits de les eines a utilitzar: multi plataforma, fàcil actualització i ràpid desplegament.
- Ambdós opcions compleixen les dues últimes premisses anotades (potencial i afinitat). No obstant estem més familiaritzats amb el desenvolupament d'aplicacions d'escriptori. Evidentment, les premisses ens marquen el camí clarament cap a l'opció de desenvolupar una aplicació web en forma d'intranet.

Una vegada decidida la plataforma en que distribuïrem i mostrarem l'aplicació de **SetelSys**, una intranet web privada, cal decidir quina tecnologia utilitzem de les existents. Les opcions que apareixen en la primera ronda de selecció són:

- Desenvolupament utilitzant **ASP .NET** a través d'una arquitectura MVC (Model – Vista - Controlador).
- Desenvolupament utilitzant Microsoft **Silverlight** a través d'una arquitectura MVVM (Model - Vista – Vista - Model).
- Desenvolupament utilitzant altres alternatives, com podrien ser l'ús combinat de **PHP + jQuery**.

D'aquesta primera ronda eliminem la tercera opció, bàsicament pels següents motius:

- Menys familiaritzats amb aquestes plataformes, i per tant el cost d'aprenentatge i formació seria molt elevat.
- Les bases de dades principals del sistema estan implementades sobre Microsoft SQL Server, i tot i que aquesta situació no implica forçosament treballar amb eines de desenvolupament Microsoft, som conscients de les eines d'integració que existeixen en la suite del Microsoft Visual Studio pel treball amb bases de dades i components del mateix fabricant.
- No sacrificuem cap potencial ni limitem les possibilitats. L'entorn de desenvolupament de Visual Studio ens brinda totes les eines necessàries per a la implementació satisfactòria del projecte SetelSys.
  - Programació orientada a objectes i sobretot a events.
  - Comunicació amb serveis web SOAP, WCF, RIA.
  - Comunicació amb bases de dades.
  - Creació de llibreries DLL
  - Creació de serveis per entorn Windows (WinServices)
  - Gestió adequada del codi i de l'entorn de depuració:
    - Tasques pendents (ToDo Lists)
    - Punts de parada (BreakPoints)
    - Gestió de marcadors
    - ...
  - I altres que no seran necessàries per aquest projecte.

Finalment hem arribat a una disjunció entre dos possibles plataformes:

- ASP .NET <sup>1</sup> (Arquitectura **MVC**) <sup>2</sup>
- Silverlight <sup>3</sup> (Arquitectura **MVVM**) <sup>4</sup>

L'escollida entre aquestes dos és el **Silverlight**, pels següents motius:

- Més orientada a l'elaboració d'una **Intranet privada**, que no pas la plataforma ASP .NET. Aquesta última està més pensada per solucions públiques que requereixin cert nivell d'obertura i publicació de la seva meta informació (sobretot pel posicionament web).

---

<sup>1</sup> Microsoft – [The official Microsoft ASP.NET site](#)

<sup>2</sup> Microsoft – [ASP MVC Overview](#)

<sup>3</sup> Microsoft – [Silverlight](#)

<sup>4</sup> MSDN – [Design Patterns - MVVM - Model View ViewModel Pattern](#)

- Permet l'elaboració **d'interfícies d'usuari més riques** i basades en els controls WPF (*Windows Presentation Foundation* <sup>5</sup>), que donen molta més gama de controls i personalització dels mateixos. Per una intranet corporativa on hem de mostrar moltes dades, graelles, gràfics representatius, etc., necessitem d'utilitzar una interfície rica i potent. És el que Microsoft anomena "*Reach vs. Rich*"<sup>6</sup>.
- Aquesta aproximació de Silverlight en l'extrem "*Rich*", ens permet desenvolupar amb molta més **agilitat i rapidesa** les **interfícies** d'usuari, sense dependre de components de terceres empreses que si que hauríem d'adquirir per aconseguir els mateixos resultats amb ASP .NET
- ASP .NET és una plataforma "*server side*", mentre que **Silverlight** és "*client side*". Això vol dir que en ASP .NET tot s'executa sobre el servidor en tot moment, mentre que el client Silverlight ho fa en el client. El gran avantatge d'aquesta característica és que, tot i tenir una càrrega inicial un pel més lenta, una vegada carregat l'script en la màquina client, la transició entre pàgines, finestres, etc., és molt més **fluida**. I si a més tenim l'aplicació guardada en la memòria cau del nostre ordinador, aleshores l'únic element depenent de la connexió seran les crides als serveis web de consulta de dades remotes.
- Estem més **familiaritzats** també en solucions basades en WPF que no en les de ASP.
- Tal i com s'explicarà en l'apartat corresponent a les eines seleccionades per a les fases de proves, la selecció del Silverlight pot comportar una sèrie de inconvenients menors a nivell d'automatització. En aquest cas, per a poder usar directament les eines d'automatització de proves que ofereix la *suite* del Visual Studio, requeriria una sèrie de modificacions en els objectes de l'aplicació. No obstant això, s'han avaluat les alternatives, optant per una solució que permet usar el Silverlight sense perdre potencial a nivell d'automatització.

---

<sup>5</sup> Microsoft Developer Network – [Windows Presentation Foundation](#)

<sup>6</sup> Tech Republic – [Create rich interfaces with Microsoft Silverlight](#)

### 3.1.2 Sistema gestor de bases de dades (SGBD) escollit

L'ERP principal disposa actualment ja d'un clúster de servidors dedicats a allotjar les aplicacions i bases de dades pròpies del seu sistema utilitzant Microsoft SQL Server com a SGBD. Pel nostre projecte SetelSys, utilitzarem també Microsoft SQL Server. En concret utilitzarem la versió **Microsoft SQL Server 2008 R2 Standard**.

#### Per què Microsoft SQL Server?

Els principals motius de la selecció del Microsoft SQL Server són les següents:

- Suficientment potent per a integrar les dades del projecte SetelSys.
- Integració més ràpida i fàcil amb l'entorn de Visual Studio .NET.
- Més facilitats a l'hora d'interactuar amb les altres bases de dades ja existents en el sistema i basades també en Microsoft SQL Server.
- Disposa de les eines necessàries per a una correcta administració de tasques programades de manteniment i/o de còpia de seguretat (SQL Server Agent).
- Només caldrà desenvolupar un component de connexió amb un motor de SGBD. El mateix que construïrem per les connexions amb la base de dades del ERP ens serviran per la base de dades de SetelSys.

#### Per què l'edició "Standard"?

De les diferents versions existents (DataCenter, Enterprise, Standard, Web, Workgroup, Express with Advanced Services, Express with Tools, Express), i sent conscients de que existeixen moltes d'altres categories on podríem analitzar les diferències entre versions, hem optat per la versió "Standard" pels següents motius, que considerem més determinants:

- **Escalabilitat:** A nivell d'escalabilitat ens permet fer ús de fins a 4 CPUs, utilitzar un màxim de 64 GB de memòria operativa, i un màxim de 524 PBs de memòria d'emmagatzematge. Altres versions de menors prestacions limiten possibles vies de creixement posteriors, sobretot limitant la memòria operativa disponible o l'espai d'emmagatzematge.
- **Disponibilitat:** Permet una configuració posterior en clúster, mentre que la resta de versions menors, no disposen d'aquesta funcionalitat. Si el client finalment optés per la configuració d'un clúster, podríem seguir utilitzant la mateixa versió de SQL Server.
- **Analysis Services:** La versió "Standard" disposa de l'eina SQL Server Analysis Services, amb eines per l'anàlisi actiu del servidor (nivell de càrrega, temps de còmput de les consultes, ...). La resta de versions menors, no la incorporen.



### 3.1.3 Software de testing

En referència al software que s'ha seleccionat per a dur a terme totes les tasques necessàries a nivell de testing, cal tenir en compte principalment quatre grans àrees de treball:

- **Gestió de les proves:** eines necessàries per a gestionar tot el que fa referència a les proves: Pla de proves, Especificació de les proves, gestió de l'execució i resultats, així com la posterior gestió dels errors que s'han localitzat.
- **Automatització:** eines usades per a la posterior automatització de les proves seleccionades, tenint en compte tant l'aplicació web com el *backend*, i les tecnologies seleccionades per a implementar-los.
- **Repositori de codi:** en aquest cas, i tal i com es veurà en el següent apartat, s'ha usat tant durant el desenvolupament com durant les proves les mateixes eines de control.

#### *Gestió de les proves:*

Actualment en el mercat podem trobar una gran varietat de programari de “*Test Management*”, que ens permeten una gestió integral de les proves. Les eines de gestió de proves s'usen per a guardar la informació sobre **com** es duran a terme les proves, planificar totes les activitats de testing, així poder obtenir informació constant i fidedigna de com evolucionen les proves.

Cadascuna de les eines aporta una sèrie de capacitats i funcionalitats, però en el nostre cas l'eina a seleccionar havia de complir els següents requeriments:

- Permetre'ns gestionar des del principi fins al final els nostres requeriments, proves i fases d'execució.
- Tenir una clara traçabilitat de tots els elements del projecte (ex: poder veure d'un requisit concret, quin és l'estat de les proves i quines incidències relacionades s'han detectat).
- Una integració senzilla amb la resta d'eines seleccionades per al desenvolupament
- I finalment, i tot i no ser un requisit indispensable inicialment, havia de permetre treballar de forma vinculada a nivell de proves automàtiques, amb les eines que es seleccionessin.

El fet que a nivell de desenvolupament haguéssim optat per a treballar amb la *suite* del Microsoft Visual Studio, ens va simplificar de forma clara la selecció de l'eina de gestió de proves. Hi ha diverses alternatives força conegudes amb les seves corresponents capacitats, però no obstant, nosaltres vam optar des del primer moment per utilitzar el **Microsoft Test Management (MTM)**, utilitzant a la vegada el **Team Foundation Server (TFS)**, bàsicament perquè el propi MTM el requereix per a desar tots els elements.

### *Automatització de proves:*

Com s'ha mencionat breument en l'apartat de selecció de les eines i tecnologies de desenvolupament, aquesta selecció té una influència directa sobre la selecció de les eines per a l'automatització de les proves.

El punt més important ha tenir en compte és el fet, que amb la selecció de Silverlight per al desenvolupament de part del nostre sistema, vam haver de prendre algunes decisions per a poder realitzar correctament l'automatització.

Una de les nostres primeres premisses era que les eines d'automatització s'integressin correctament i fàcilment amb la resta d'eines de desenvolupament, fet que ens va portar ràpidament a utilitzar el mateix Visual Studio, concretament els que anomenen **Coded UI Tests**, tests automàtics per a realitzar proves usant la seva interfície, és a dir, proves funcionals que utilitzen de forma directa tots els elements del sistema, tal i com veurem més en detall en l'apartat específic de [proves automàtiques](#).

Per a la part del *backend* s'ha optat directament per aquesta solució, ja que s'integrava correctament i sense problemes.

No obstant això, segons la pròpia documentació de Microsoft, per a poder reconèixer i interactuar amb els diferents objectes de l'aplicació codificada usant Silverlight, cal realitzar primer de tot una sèrie de modificacions en el codi:

- Caldria afegir la referència  
Microsoft.VisualStudio.TestTools.UITest.Extension.SilverlightUIAutomationHelper.dll a l'aplicació per tal de que els controls Silverlight es poguessin identificar
- A més, aquesta *dll* caldria afegir-la de forma condicional i no distribuir-la amb la resta de la solució.<sup>7</sup>
- I amb tots aquests canvis, en molts dels casos, es segueixen tenint dificultats a l'hora de poder reproduir correctament les proves.

Degut a aquestes circumstàncies, vam optar per utilitzar una eina diferent per al *frontend*, una eina que per experiències laborals anteriors requeria de molt pocs passos per a poder detectar els objectes de Silverlight senzillament. L'eina seleccionada va el **TestComplete**, de l'empresa SmartBear. Test Complete és una eina purament d'automatització, per a gestionar la creació, el manteniment i la posterior execució de les proves, tant per aplicacions d'escriptori, web com mòbil. En l'apartat corresponent veurem en detall el funcionament d'aquesta aplicació, i les proves automatitzades, però per a poder interactuar amb l'aplicació únicament ens calia realitzar un *patch* del fitxer .xap en el servidor, i d'aquesta manera l'aplicació ja era capaç de reconèixer i interactuar amb els objectes.

---

<sup>7</sup> [Testing Silverlight Applications with Coded UI Tests or Action Recordings](#)

La gràcia de seleccionar aquesta eina és que, a part de la facilitat amb la que ens permet interactuar amb l'aplicació, té una integració molt bona amb el Team Foundation Server i la suite de Visual Studio, fet que ens ha permès lligar correctament les proves dissenyades al Test Manager, amb les seves corresponents proves automàtiques.<sup>8</sup>

### *Repositori de codi:*

Sense necessitat d'entrar en molt de detall, i tal i com s'explica a continuació, a nivell de testing també s'ha usat les mateixes eines de control de versions per a gestionar el codi desenvolupat durant la fase de proves automàtiques.

---

<sup>8</sup> SmartBear – [Integration With Microsoft Visual Studio](#)

### 3.1.4 Treball en equip i desenvolupament compartit

Per tal de poder realitzar conjuntament les tasques de desenvolupament, així com les de les fases de proves, i durant la correcció d'errors, hem fet ús d'una eina de control de versions, utilitzant un client de l'**Apache Subversion (SVN)** <sup>9</sup>, anomenat **TortoiseSVN**. <sup>10</sup>

Aquesta aplicació ens permet sincronitzar el codi de forma remota i controlar els canvis generats en cada nova versió que es penja al **repositori** SVN. Els motius de l'elecció de TortoiseSVN:

- Ens permet integrar un **servidor Apache propi** en un equip servidor intern a la nostra xarxa de treball i amb accés remot a través de les corresponents configuracions, per a poder disposar d'una **eina gratuïta** (ja sigui per un ús comercial o no) sense cap cost per nosaltres.
- Disposa de les **funcionalitats** necessàries pel nostre projecte:
  - Descarregar i mantenir actualitzada una rèplica del repositori en el nostre equip.
  - Integració amb l'explorador de Windows (ràpidament podem visualitzar si un arxiu està actualitzat a la última versió o no a través de la icona de l'arxiu).
  - Capacitat d'explorar el contingut del repositori.
  - Ens permet descarregar registres de tots els canvis existents abans de pujar una nova versió al repositori.
  - Ràpida i còmoda actualització dels nostres arxius locals, ja sigui per descarregar la última versió (Update) o bé per a penjar tots els canvis que hem fet (Checkout).
- A través del complement **AnkhSVN** <sup>11</sup>, que instal·lem dins l'entorn del Visual Studio, podem visualitzar tota la informació referent al sistema de subversions, i per tant tenim accés constant i immediat a la seva informació (última versió, última data i hora de modificació, ...)

---

<sup>9</sup> [Apache Subversion](#)

<sup>10</sup> Tortoise SVN - [Home](#)

<sup>11</sup> AnkhSVN – [Subversion Support for Visual Studio](#)

### 3.2 Selecció de la plataforma hardware per allotjar l'aplicació

Per poder evitar l'ús de varis servidors amb processos i funcionalitats disperses, i no sobresaturar els servidors dedicats exclusivament a les connexions dels terminals i accés a les bases de dades de l'ERP Principal és necessari un nou servidor que ens permeti:


- Unificar els servidors actuals (justos de recursos i obsolets a nivell de plataforma).
- S'especialitzi i es dediqui a servir peticions del sistema satèl·lit del SetelSys, ja sigui a través de la seva aplicació web hostatjada en un servidor IIS o bé per l'aplicació de manteniment del sistema.
- Permeti una execució controlada i autònoma dels processos automatitzats necessaris.
- Disposi de l'estabilitat i fiabilitat necessària, així com de la potència suficient per a dur a terme aquestes múltiples tasques de manera eficient i paral·lela.

Actualment l'empresa disposa d'una sala de servidors amb un armari on resideixen els diferents elements centrals del sistema hardware (servidors, SAls, commutadors de xarxa, routers,...). Aprofitarem aquesta estructura i eix neuràlgic per integrar el servidor dedicat a les tasques del sistema SetelSys, en format de rack 2U, per poder-lo connectar també amb el sistema d'alimentació ininterrompuda i deixar-lo integrat en l'armari principal.

Plantegem doncs l'adquisició del següent servidor i components necessaris per la seva posada en marxa i operativitat:

<p><b>x1</b></p> <p>Servidor HP ProLiant DL360P Gen8          Processador Xeon E5-2620 a 2.00GHz          8 GB de memòria RAM ECC          Controladora SAS HP Smart Array P420i          DVD-RW          Format rack 2U</p>	
<p><b>x3</b></p> <p>Discs durs de 500 GB Gen8 6GB SAS          HotPlug (connexió en calent)          15.000rpm          2.5"          Configurats en RAID 5</p>	
<p><b>x1</b></p> <p>Microsoft Windows Server 2008          Versió "Standar".</p>	
<p><b>x5</b></p> <p>Llicències d'accés de client (CAL)          Incorporades amb el Windows Server 2008.</p>	

El client també ha optat per incorporar els següents serveis i components opcionals oferts:

<b>x1</b> Extensió de garantia HP CarePack 3 anys de garantia amb resposta de servei l'endemà de l'obertura de la incidència.	
<b>x1</b> Mòdul extra de memòria RAM ECC de 8 GB El sistema queda aleshores amb 16 GB de memòria RAM total.	
<b>x1</b> Font d'alimentació redundat HP Permet mantenir l'equip en funcionament en cas de la fallida de la font d'alimentació principal.	

### *Possibles millores descartades en la plataforma de hardware*

També s'ha plantejat al client l'opció d'instal·lar dos servidors amb la mateixa configuració que la mencionada en les anteriors taules amb un sistema de "clúster". Una configuració en "clúster" ens permetria que el sistema reaccionés transparentment si es produeix la caiguda d'un dels servidors.

En el moment que un dels servidors queda fora de servei pel motiu que sigui, l'altre servidor "redundant" del clúster, s'encarrega de la gestió del sistema i per tant evita la paralització dels processos de l'empresa.

No obstant aquesta opció ha estat desestimada inicialment pels costos duplicats a nivell de hardware que això suposa.

### 3.3 Arquitectura de la solució

Per tal de poder desplegar de forma eficient i controlada el sistema de gestió del SetelSys, el sistema estarà conformat per:

- **SetelSys Server:** Equip servidor, encarregat d'allotjar les aplicacions, serveis web, serveis de Windows i les bases de dades pròpies del SetelSys, amb la configuració de hardware anotada en l'anterior apartat.
- **Internet Information Services (IIS)** instal·lat i configurat com a rol en dit servidor i encarregat de subministrar i implementar l'accés a l'aplicació web de SetelSys, així com als serveis web necessaris per la interacció amb les bases de dades.
- **SetelSys:** Aplicació web "*Front End*" desenvolupada amb Silverlight v4: Es podria haver desenvolupat amb Silverlight 5, però degut a que certs equips clients treballen amb sistemes operatius bastant antics, i que la versió Silverlight té alguns problemes de compatibilitat amb aquestes versions, hem optat per desenvolupar-ho amb la SDK v4 del Silverlight.
- **SetelSys Core:** Nucli intern de la solució, encapsulat en una llibreria de Windows (DLL), amb totes les funcionalitats necessàries per l'execució de la solució:
  - Capa de gestió i control d'execució i registre d'errors (Midgard)
  - Classes d'accés als diferents entorns de dades:
    - WebDataCom
    - SetelSysDataCom
    - ERPDataCom
  - Classe per l'accés controlat a bases de dades SQL Server, ja sigui en execucions simples o de múltiples comandes, i també poden escollir l'execució dins de transaccions.
  - I la resta de classes que analitzarem més en profunditat en l'apartat del desenvolupament.
- **SetelSys Crono:** Servei de Microsoft Windows encarregat d'executar tots els processos automatitzats segons la planificació especificada en la seva configuració.
- **Maiden:** Servei web WCF per a resoldre peticions dels clients de l'aplicació web SetelSys i el servidor de l'aplicació web que a la vegada actua com a servidor de bases de dades.
- **SetelSys Runner:** Programari client "*Back End*" desenvolupat amb WinForms utilitzant la versió 4 de .NET Framework. Aquest programari s'utilitza per operacions de manteniment i control bàsiques, tant pel que fa als paràmetres del sistema com a l'observació de l'execució de processos automatitzats i la configuració de la seva planificació, entre d'altres funcionalitats que veurem més endavant.

Cal anotar que el servidor web de l'aplicació està hostatjat en la mateixa sala de servidors des d'on té accés per LAN a les diferents bases de dades allotjades en els servidors de l'ERP principal.

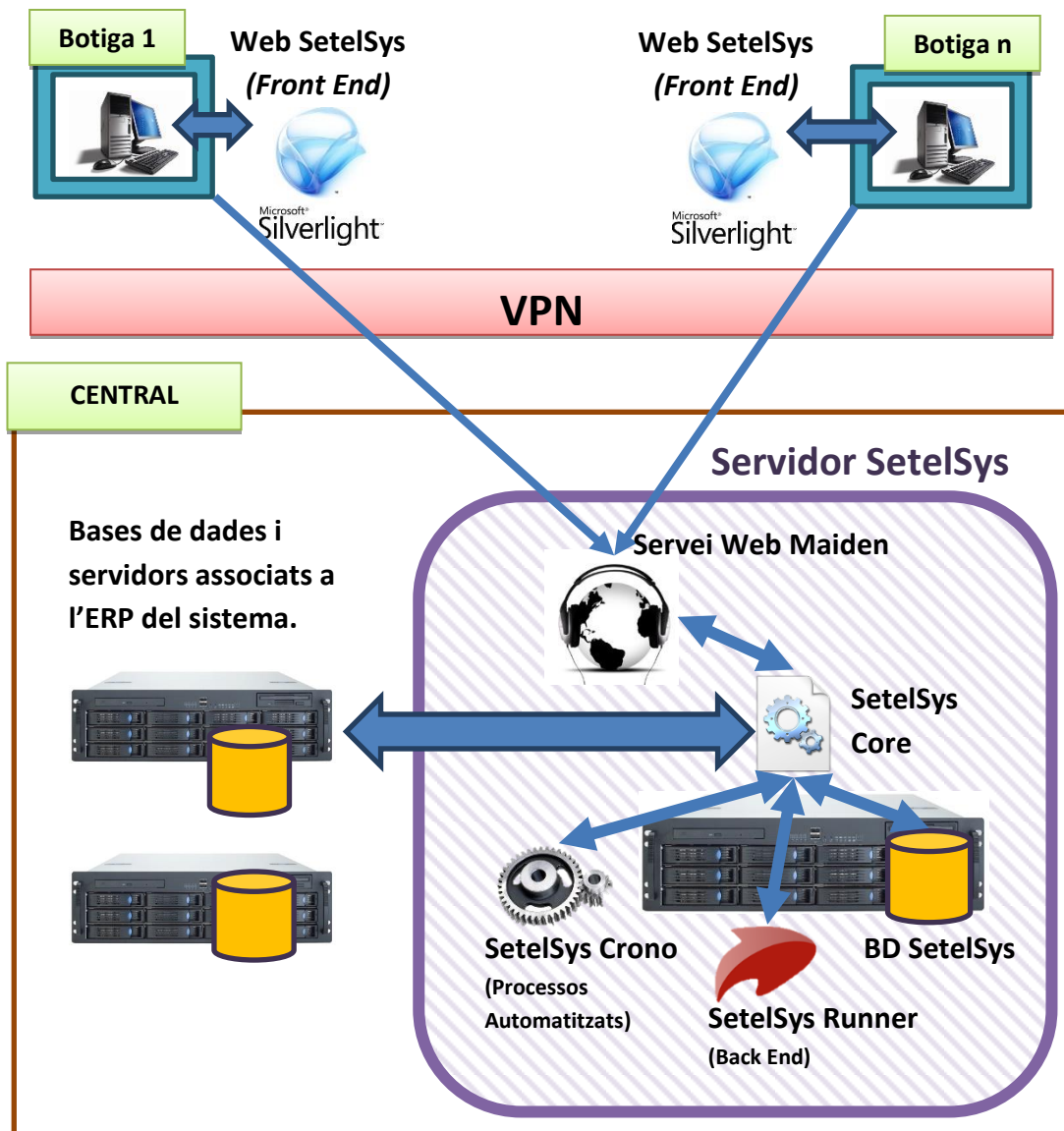
Totes les connexions remotes que s'estableixen cap al servidor de l'aplicació web del SetelSys són des de les diferents botigues cap a la central (on resideixen els servidors) a través d'una VPN establerta per hardware amb encaminadors Cisco, utilitzant una xarxa NetLan de Movistar.

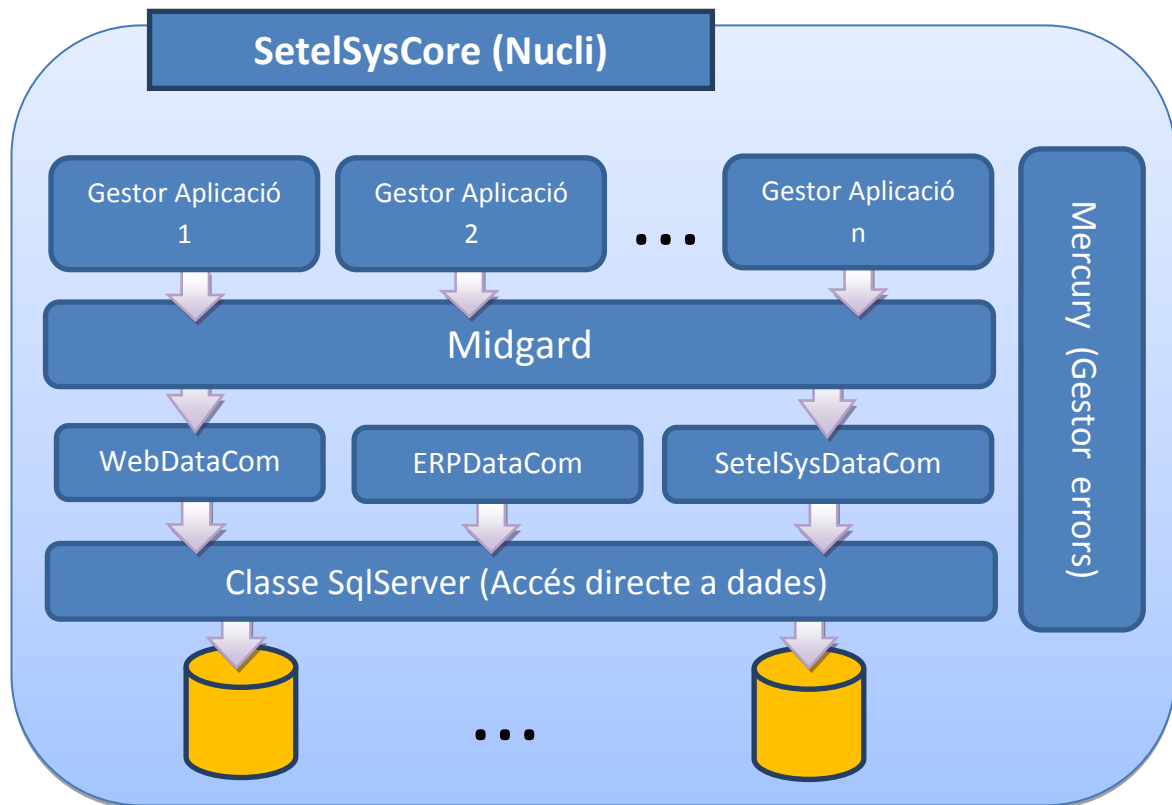
A través d'aquesta VPN les dades viatgen sempre sobre un canal segur i encriptat, fent innecessària la implementació d'un certificat SSL per al desplegament de l'aplicació web del SetelSys.

També s'han creat certes aplicacions d'ajuda en el control del sistema i per a facilitar la instal·lació d'actualitzacions:

- **DB Comparer** → Comparador d'esquemes de bases de dades.
- **FTP Deployer** → Per la instal·lació d'actualitzacions de SetelSys en el servidor de producció.
- **ChronoChecker** → Per a la visualització de l'estat del servei de SetelSys.
- **DB Tester** → Validació de la connexió en les diferents bases de dades involucrades en el procés.



*Diagrama arquitectònic de la solució*

*Diagrama arquitectònic del nucli (SetelSysCore)*

En l'apartat corresponent al [desenvolupament del nucli](#), entrarem més en detall en l'objectiu i el funcionament intern de les diferents capes del nucli.

### 3.4 Salvaguarda de les dades i del servidor

Hem trobat oportú dedicar el temps necessari en planificar l'estructura de les còpies de seguretat de les dades i de la configuració inicial del servidor.

#### Backup de les bases de dades de l'ERP

El sistema ERP principal, disposa dels seus corresponents servidors en clúster i d'un sistema de còpies automatitzat, tant pel que fa a les dades d'arxius d'usuari com de les bases de dades. En aquest punt per tant no cal incidir-hi ja que està corresponentment cobert i resolt.

#### Backup de les bases de dades de SetelSys

Pel que fa a les bases de dades generades per l'ús de la nostra aplicació SetelSys, hem planificat el següent sistema:

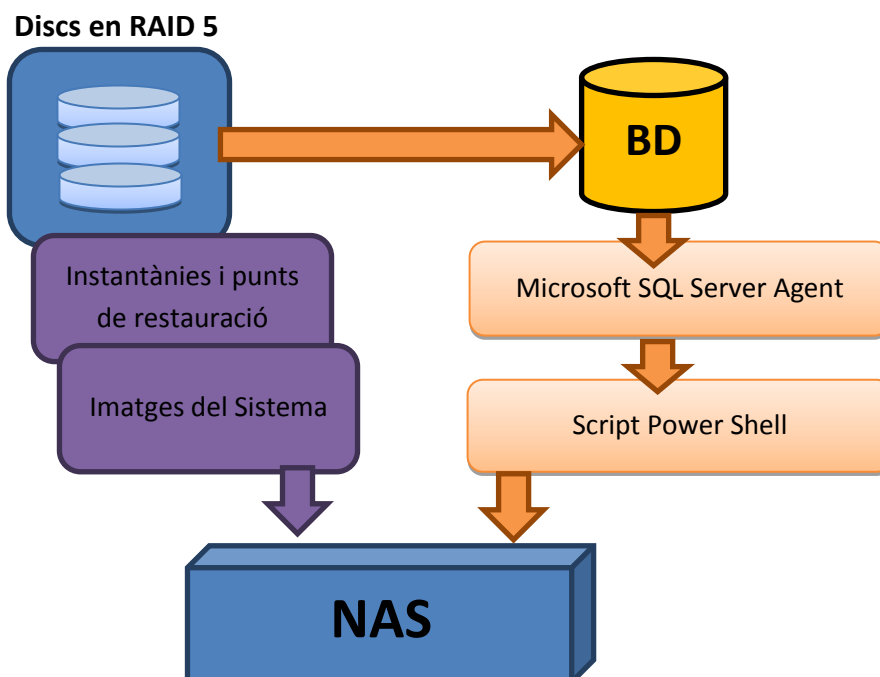
- **Execució de la còpia de seguretat:** Es planifica una tasca utilitzant l'agent del Microsoft SQL Server (SQL Server Agent) que generi una còpia de forma diària a les 12:00h A.M. i la desi en un arxiu localitzat del disc dur amb el nom "BACKUP-dd-mm-yyyy-hh-mm.bak" (dd=dia, mm=mes, yyyy=any, hh=hora, mm=minut).
- **Emmagatzematge de la còpia:** A les 01:00h A.M. s'executa una tasca programada en el servidor (i preparada amb les comandes de Microsoft Power Shell) de forma que guardi l'arxiu generat en el punt anterior en un equip NAS (Network Area Storage). L'entorn de xarxa és Gigabit, amb la qual cosa aquest procés és força ràpid.
- **Neteja de còpies obsoletes:** Aquesta mateixa tasca programada, en el cas que el punt anterior s'executi correctament, s'encarrega d'eliminar còpies velles per evitar un consum innecessari d'espai d'emmagatzematge. Totes les còpies anteriors a 7 dies, són purgades de forma automàtica per aquest script.

### Backup del sistema

Els discos durs del servidor, configurats en un sistema RAID 5, disposen d'un sistema propi de còpia donada l'arquitectura i nivell del sistema RAID. En el cas de l'errada d'algun dels discos, podríem recuperar la resta de les dades amb els existents regenerant el sistema RAID amb els que estarien encara funcionals.

No obstant, per maximitzar la seguretat i salvaguarda de l'equip servidor, hem configurat:

- **Sistema d'instàntànies i punts de restauració** per poder, en qualsevol moment, fer un *rollback* del sistema i tornar al punt anterior.
- Planifiquem de **forma trimestral**, una parada de manteniment que dediquem, entre d'altres punts a revisar, a generar una **imatge exacta del sistema** per poder-la restaurar en cas d'una fallida que no es pogués recuperar per cap de les dos vies anteriors. Tenint en compte que el servidor, a nivell d'aplicacions, no és molt dinàmic (o és sobretot a nivell de bases de dades i aquestes ja queden cobertes segons el protocol establert en l'anterior punt), aquest interval de temps és més que suficient per assegurar la disponibilitat de qualsevol recuperació.



### 3.5 Convenis i nomenclatura utilitzada per a la codificació

De cara a utilitzar una nomenclatura estandaritzada en el procés de desenvolupament, hem marcat un seguit de premises a seguir en compte durant la codificació d'interfícies, classes, propietats i mètodes. Aquesta nomenclatura establerta no només s'ha usat durant les fases de desenvolupament, sinó que s'ha establert també com a nomenclatura per a les fases de proves, especialment durant l'automatització. A més, tal i com s'explica en l'apartat corresponent, també s'ha establert una sèrie de paràmetres dins la nomenclatura per a les funcions de test.

#### 3.5.1 Conveni global

De forma general, aplicarem una normativa global al moment d'escollir el nom d'un control, classe, enumeració, propietats, ... Caldrà complir les següents normes:

- No s'utilitzaran espais
- No s'utilitzaran símbols d'accentuació.
- No s'utilitzaran guions, comes, punts,...
- No s'utilitzaran caràcters com "ç", ni apòstrofs, ...
- Si el nom compren més d'una paraula, la primera lletra s'anotará sempre en majúscula.

Buscarem sempre un nom adequat per identificar-ho, excepte els components globals encarregats de l'arquitectura funcional, els quals utilitzarem noms en clau i breus (Heimdall, Mercury, Seshat, ERPDataCom, Midgard,...)

Exemples: *ConsultarModelsAssignats*, *Cancel·larFinestra*, *NomUsuari*, *LlistaMagatzems*, *ConsultarUnitatsPendants*, *ModificarDocumentVenda*, *EliminarCaixaMagatzem*,...

### 3.5.2 Noms dels controls gràfics de les interfícies d'usuari (UI)

El nom que anotarem als controls estarà format per un prefix i un sufix complint les següents característiques.

**Prefix:** Utilitzarem 3-4 lletres inicials per a definir la classe del control, i ser més fàcil així la seva localització per codi. Exemples:

Sufix	Classes de controls on utilitzar-lo
<b>txt</b>	TextBox, TextBlock
<b>cmd</b>	Button
<b>opt</b>	OptionButton, RadioButton
<b>chk</b>	CheckBox
<b>grid</b>	DataGrid, DataGridView
<b>lbl</b>	Label
<b>cmb</b>	ComboBox
<b>grp</b>	GroupBox
<b>frm</b>	Form
<b>win</b>	ChildWindow
<b>tab</b>	TabCollection
<b>lst</b>	ListBox, ListView
...	...

**Sufix:** Utilitzarem la normativa global.

Aplicant les normes vistes pel prefix i sufix, alguns possibles exemples complets de nom de control podrien ser: *gridAssignarModel*, *cmdCancelar*, *txtNomUsuari*, *lstLlistaMagatzems*, *cmdConsultarUnitatsPendants*,...

### 3.5.3 Nom de les classes

Utilitzarem la normativa global a excepció de les classes representatives dels objectes de bases de dades, en els quals utilitzarem el prefix especial "DB\_" per que quedi sempre constància de que treballem amb objectes que encapsulen les propietats de taules de les bases de dades.

Exemples: *GestorConfiguracioTarifes*, *ParametresPrevisioTraspasos*, *ErrorSistema*,...

### 3.5.4 Nom de les propietats de les classes

En aquest cas utilitzarem dos criteris segons la naturalesa de la propietat:

#### Propietats d'àmbit públic

Utilitzarem la normativa global.

Exemples: *NomUsuari*, *CodiTarifa*, *DataCompra*,...

#### Propietats d'àmbit privat

Sempre les anotarem amb MAJÚSCULA, i utilitzant com a primer caràcter i separador de paraules un guió baix ("\_"). L'objectiu és diferenciar clarament les propietats públiques de les privades.

Exemples: *\_ID\_COMPRA*, *\_NOM\_TARIFA*, *\_CODI\_ARTICLE*

### 3.5.5 Nom dels mètodes i funcions

Utilitzarem la normativa global i posarem especial atenció a denotar una semàntica adequada amb les paraules utilitzades per ajudar a la identificació de la seva funció.

#### Prefixos CRUD

Per anomenar els mètodes o les funcions que es dediquin a l'operativa CRUD (Create – Read – Update – Delete) intentarem sempre utilitzar un mateix prefix.

Taula de prefixos a utilitzar en operatives CRUD:

Operativa	Prefix a utilitzar	Exemples
<b>CREATE</b>	Afegir	AfegirNouUsuariSetelSys(...)
<b>READ</b>	Consultar	ConsultarUsuarisActius()
<b>UPDATE</b>	Modificar	ModificarUsuari(...)
<b>DELETE</b>	Eliminar	EliminarUsuari(...)

#### Atributs dels mètodes i funcions

En els atributs que assignem a la definició d'un mètode o funció, utilitzarem la normativa global. No obstant intentarem definir sempre que sigui possible o interessant (segons criteri del desenvolupador), el sumari de l'element, anotant la descripció de l'ús del mètode o funció així com dels seus paràmetres per permetre una lectura còmoda i intel·ligible en les crides posteriors des d'altres classes.

### 3.5.6 Nom dels delegats, predicats, ...

En el cas d'utilitzar classes meta estructurals com els delegats, predicats i altres, utilitzarem un prefix inicial per a denotar el tipus de representació.

Exemples: *predFiltrarFiles*, *delCarregarInformacio*, ...

### 3.5.7 Resta d'elements de desenvolupament

Per la resta (enumeracions, events, variables, ...) utilitzarem la normativa global.

### 3.5.8 Funcions de test

Les funcions que s'utilitzen a l'hora de codificar els tests, que en el Visual Studio s'identifiquen inicialment amb el *tag* [Test Method], a més a més de complir amb la resta de nomenclatura estàndard que s'han explicat en els apartats anteriors, és important remarcar que s'ha optat per a donar noms molt exemplificadors i identificatius, per a tenir clar quina és la prova que s'està duent a terme en aquella funció.

L'estratègia de proves seleccionada permet que les proves ens si mateixes siguin força *auto-documentades*, en el sentit que són senzilles de seguir i de llegir, i amb la selecció d'un nom clar permet ràpidament entendre les proves a fer.

Alguns dels exemples poden ser:

- *CreacióNouUsuariAcces*: provar l'accés a la funcionalitat de creació de nou usuari.
- *CreacióNouUsuariNomObligatori*: prova per a comprovar si la validació de que el nom de l'usuari és obligatori es compleix.

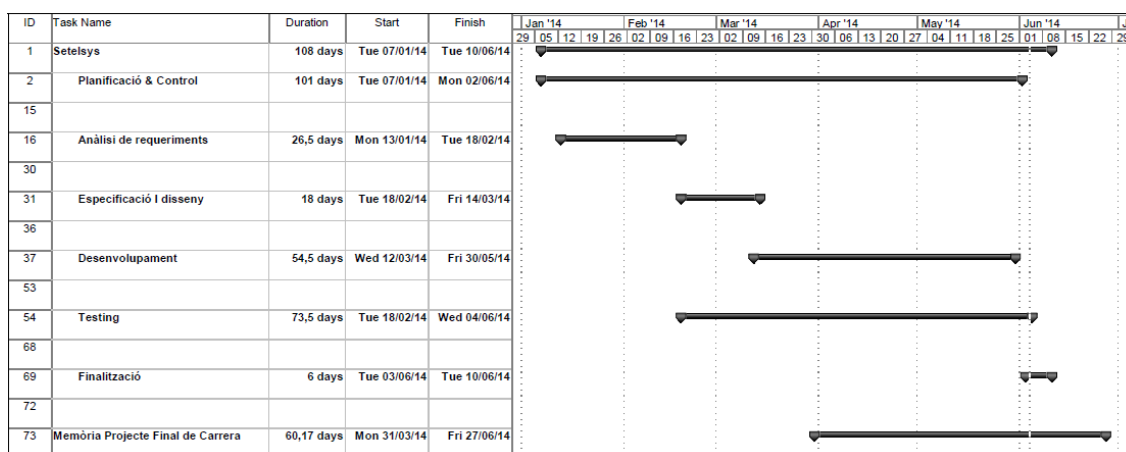


## 4. Planificació

A continuació es presenta la planificació general duta a terme al principi del projecte, i posteriorment s'expandeixen els diferents apartats per a veure en detall les diferents tasques que componen cadascuna de les seccions en les que s'ha dividit el projecte.

### 4.1 Diagrama de Gantt inicial

La versió esquemàtica de la planificació inicial que es va dur a terme, en forma de diagrama de Gantt és la següent:



Tal i com es pot veure, a grans trets s'ha separat per un costat el projecte SetelSys en sí mateix, de la creació de la documentació per al projecte de final de carrera, i dins del projecte SetelSys, s'ha estructurat de la següent forma:

- Planificació i Control
- Anàlisi de Requeriments
- Especificació i Disseny
- Desenvolupament
- Testing
- Finalització

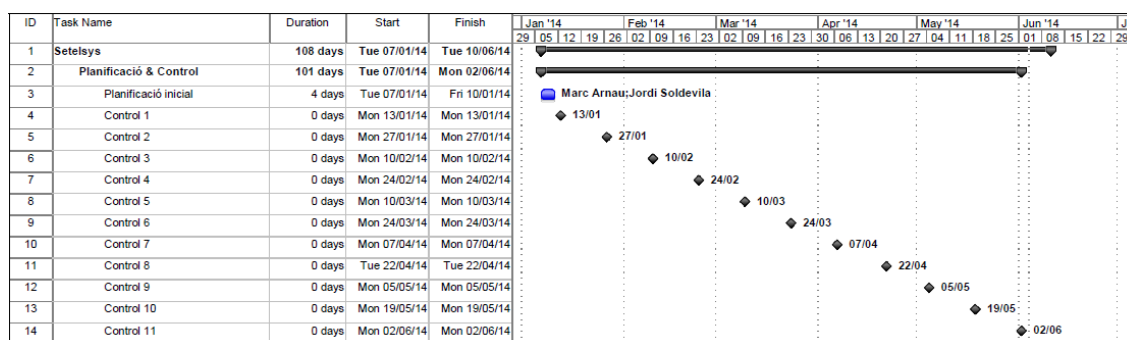
En el següent apartat és detallen cadascun dels apartats, i s'expliquen breument les tasques en que s'han subdividit el projecte, però a nivell general cal remarcar que s'ha optat per treballar de forma conjunta, amb una sèrie de punts de control (tant a nivell de SetelSys com a nivell de la creació de la memòria del projecte) on hem anat compartint els avanços en les diferents tasques compartides, i a la vegada, hem han compartint el detall de les tasques més particulars de cadascú, principalment en les àrees de Desenvolupament i Testing.

També s'ha realitzat un anàlisi econòmic de la solució plantejada, però s'ha optat per a explicar-lo en un [apartat](#) al final de la memòria.

## 4.2 Descripció de les tasques

### 4.2.1 Planificació i Control

El diagrama de Gantt en detall de la fase de **Planificació i Control** és el següent:

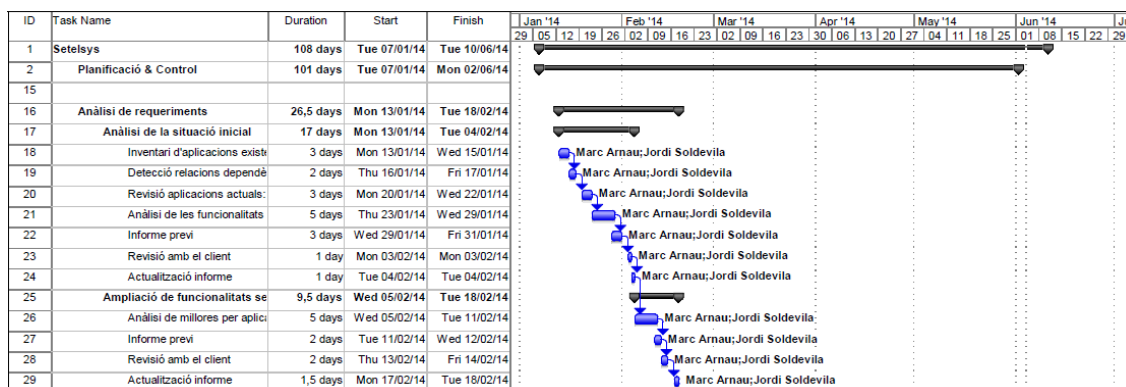


Les seves tasques consisteixen en:

- **Planificació inicial:** tasca integrament dedicada a la planificació de tot el projecte, tenint com a resultats aquests diagrames de Gantt que s'estan analitzant.
- **Punts de control:** al fer un treball en equip, i tot i la interacció força constant, es va optar per a marcar una sèries de fites de control per a revisar planificació, tasques en progrés tant comunes com específiques, modificacions, etc...

## 4.2.2 Anàlisi de Requeriments

El diagrama de Gantt en detall de la fase de **Planificació i Control** és el següent:

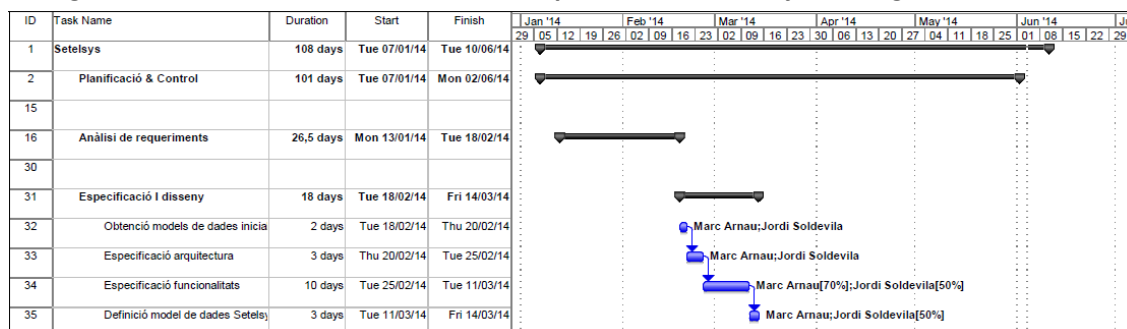


Les seves tasques consisteixen en:

- Anàlisi de la situació inicial
  - **Inventari d'aplicacions existents:** tal i com s'explica en el "Punt de Partida", no es tractava d'un sistema sorgit des de zero, sinó que existia un sistema vigent amb el que s'havia de treballar, i per tant calia inventariar que hi havia.
  - **Detecció relacions dependència entre aplicacions:** un cop inventariades les aplicacions, calia veure quines dependències hi havia entre elles
  - **Revisió aplicacions actuals:** completa i vigència: registrar per a totes les aplicacions existents, l'estat en el que estaven, l'ús que se'n feia així com si calia seguir mantenint-les.
  - **Anàlisi de les funcionalitats actuals:** a part del inventari d'aplicacions, calia veure totes les funcionalitats que es trobaven en el sistema inicial.
  - **Informe previ:** documentació sobre les anteriors tasques d'anàlisi de la situació actual
  - **Revisió amb el client:** revisió del document sorgit de la tasca anterior amb el client per a detectar errors, mancances, millores, etc.
  - **Actualització informe:** processament dels canvis suggerits pel client.
- Ampliació de funcionalitats segons requisits del client
  - **Anàlisi de millores per aplicació:** analitzar totes les millores demanades per part del client.
  - **Informe previ:** documentació sobre les millores a implementar a l'aplicació.
  - **Revisió amb el client:** revisió del document sorgit de la tasca anterior amb el client per a detectar errors, mancances, millores, etc.
  - **Actualització informe:** processament dels canvis suggerits pel client.

### 4.2.3 Especificació i Disseny

El diagrama de Gantt en detall de la fase de **Especificació i Disseny** és el següent:

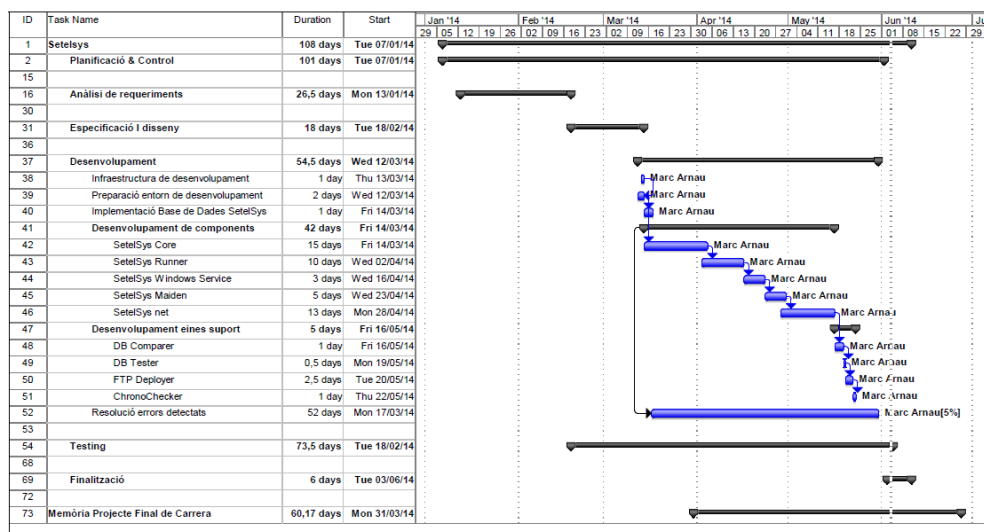


Les seves tasques consisteixen en:

- **Obtenció models de dades inicials:** procés d'enginyeria inversa per a obtenir els models de dades, de les bases de dades inicials en el sistema.
- **Especificació arquitectura:** detall de l'arquitectura de components del sistema, així com les dependències entre els diferents elements.
- **Especificació funcionalitats:** especificació en detall de totes les funcionalitats a implementar a SetelSys.
- **Definició model de dades SetelSys:** especificació del nou model de dades per al nou sistema.

## 4.2.4 Desenvolupament

El diagrama de Gantt en detall de la fase de **Desenvolupament** és el següent:

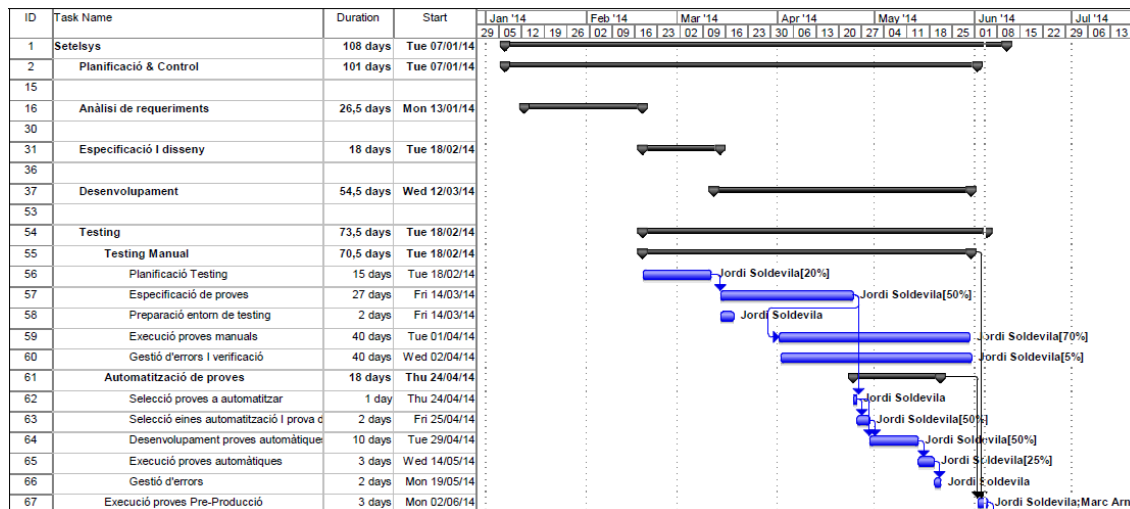


Les seves tasques consisteixen en:

- **Infraestructura de desenvolupament:** instal·lació i validació de les eines i entorn de desenvolupament, necessaris per a dur a terme les tasques posteriors.
- **Preparació entorn de desenvolupament:** posada en marxa del que serà l'entorn de desenvolupament, on s'aniran instal·lant i preparant els diferents elements que s'aniran desenvolupant posteriorment.
- **Implementació Base de Dades SetelSys:** desenvolupar la nova base de dades per al sistema SetelSys segons les especificacions realitzades durant la fase anterior.
- **Desenvolupament de components:** desenvolupament dels diferents components, com són:
  - SetelSys Core
  - SetelSys Runner
  - SetelSys Windows Service
  - SetelSys Maiden
  - SetelSys net
- **Desenvolupament eines suport:** desenvolupament de les eines que suport que s'usaran en el sistam, i que són les següents:
  - DB Comparer
  - DB Tester
  - FTP Deployer
  - ChronoChecker
- **Resolució errors detectats:** correcció dels errors i incidències detectats, tant durant la fase de desenvolupament mateix, com durant tot el procés de proves de software que s'anirà duent a terme.

## 4.2.5 Testing

El diagrama de Gantt en detall de la fase de **Testing i Automatització** és el següent:

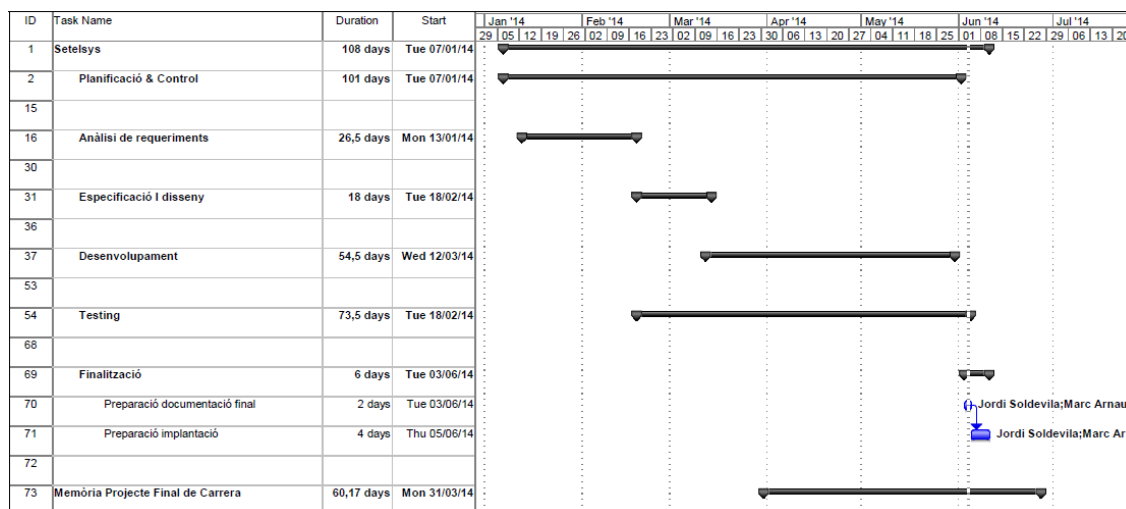


Les seves tasques consisteixen en:

- **Testing Manual**
  - **Planificació Testing:** seguint la metodologia escollida (TMap), fase de planificació, i preparació del pla de proves. Inclou la posada en marxa de l'eina de gestió de proves.
  - **Especificació de proves:** fases de preparació i posterior especificació de les diferents proves a executar durant la posterior fase. Preparació entorn de testing
  - **Execució proves manuals:** execució de les proves prèviament desenvolupades.
  - **Gestió d'errors i verificació:** registre de totes les incidències, errors i millores detectades, i una vegada corregides i entregades, posterior validació, i en alguns casos, re-execució de part de les proves.
- **Automatització de proves**
  - **Selecció proves a automatitzar:** de totes les proves manuals prèviament especificades, realitzar la selecció segons possibilitats i prioritats, de les que seran susceptibles de ser automatitzades.
  - **Selecció eines automatització i prova de concepte:** proves de concepte, per a veure que les eines seleccionades realment funcionen correctament i el projecte d'automatització pot tirar endavant.
  - **Desenvolupament proves automàtiques:** implementació de les proves automatitzades per mitjà de les eines seleccionades.
  - **Execució proves automàtiques:** execució de les proves, per a comprovar-ne la seva validesa, i obtenir els resultats de la qualitat de l'aplicació.
  - **Gestió d'errors:** gestió de qualsevol error detectat, tant a l'aplicació SetelSys, com en les proves automatitzades.
- **Execució proves pre-Producció:** execució de les últimes proves en un entorn de pre-producció per a realitzar les últimes validacions.

## 4.2.6 Finalització

El diagrama de Gantt en detall de la fase de **Finalització** és el següent:

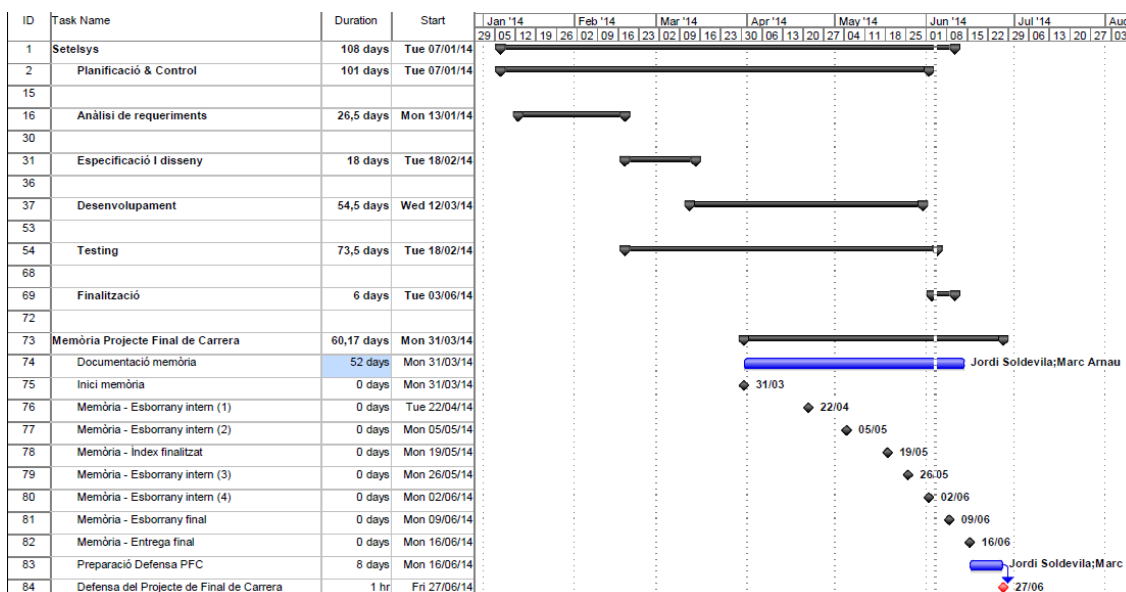


Les seves tasques consisteixen en:

- **Preparació documentació final:** generació de tota la documentació necessària, a tots els àmbits.
- **Preparació implantació:** realització de tots els passos per a dur a terme la implementació.

## 4.2.7 Memòria del Projecte de Final de Carrera

A més a més del propi projecte, ja vam planificar des del inici com duríem a terme la creació de la documentació per al Projecte de Final de Carrera. La planificació plantejada inicialment era:



### 4.3 Projecte

Per a l'enregistrament de les hores destinades en les diferents etapes del projecte, utilitzarem una eina disponible de forma gratuïta a la xarxa que ens permetrà utilitzar un espai de treball conjunt entre nosaltres on assignarem diferents projectes associats a les diferents etapes del nostre projecte global.

D'aquesta manera podrem comptabilitzar de forma còmoda i compartida el temps real destinat a l'elaboració de les diferents etapes i poder contrastar possibles desviacions finals en la finalització del projecte.

L'eina en qüestió que utilitzarem és **"Toggl"**<sup>12</sup>. És una solució web on-line que permet:

- Crear espais de treball compartits entre usuaris.
- Crear clients
- Crear projectes
- Associar projectes a clients
- Anotar tasques dedicades a un projecte, anotant-ne el temps destinat i dia, i una descripció de la tasca.
- Utilitzar el seu propi temporitzador per iniciar, continuar o parar el càlcul de temps.
- Generar informes mensuals, setmanals, entre dates, etc, on apareix un resum dels temps dedicat a cada projecte i podent visualitzar el detall de les tasques dutes a terme.

També disposen d'una versió disponible per Escriptori i per terminals mòbils (Android i iOS).

---

<sup>12</sup> Pàgina web Toggl: [www.toggl.com](http://www.toggl.com)



## 5. Anàlisi de requeriments


Per a poder desenvolupar un correcte anàlisi de requeriments, i veient la situació actual del sistema (aplicacions obsoletes, redundants, desconeixement d'ús actiu,...) haurem de seguir les següents etapes per poder concloure un anàlisi adequat a la realitat i a les necessitats actuals de l'empresa i el seu sistema satèl·lit actual:

1. Inventariar totes les aplicacions existents.
2. Definir les relacions existents entre les diferents aplicacions.
3. Determinar la completessa i vigència de les diferents aplicacions. Cal determinar quines són completes funcionalment i vigents pel que fa a la utilització actual en el sistema.
4. Analitzar la funcionalitat actual de les aplicacions, ja que en molts casos els propis usuaris desconeixen el patró de funcionament i les variables que es tenen en compte en cada aplicació
5. Contrastar les funcionalitats actuals amb el client.
6. Complementar les funcionalitats actuals amb les noves desitjades pel client.
7. Elaborar un primer informe amb el contingut dels diferents punts i sotmetre'l a revisió amb el client fins determinar, amb la màxima fidelitat possible, la llista de requeriments finals del sistema SetelSys.



### 5.1 Inventari d'aplicacions satèl·lit

Cal en primer lloc elaborar un inventari de les aplicacions existents. Anotem a continuació el resultat de l'inventari agrupat pels blocs que ja hem presentat anteriorment.

Per cada aplicació de Microsoft Access present, anotem:

- Nom descriptiu (S'anota en castellà per ser el nom en que es coneix dins l'empresa i evitar confusions a l'hora d'identificar l'aplicació)
- Breu descripció textual
- Ruta d'accés en xarxa a l'arxiu
- En el cas de que l'aplicació s'executi com un procés automatitzat, ho indiquem amb la icona 

### 5.1.1 Bloc 1: Logística


Nom	Breu descripció
<b>Datos</b>	<b>Actualitza les factures</b> de comandes ja enviades amb la informació referent al número d'expedició de l'albarà associat.
<b>Expedición Seur</b>	<a href="\\proliant\c\datos seur\Datos expedicion seur.mdb">\\proliant\c\datos seur\Datos expedicion seur.mdb</a>
 <b>Pedidos</b>	<b>Genera</b> els traspasos entre les diferents botigues i també el magatzem central, per fer-los-hi arribar el material necessari per poder servir les seves comandes de clients que estan pendents de rebre material <b>en estoc</b> . <a href="\\fileserver\...\Pedidos tiendas\pedidos.mdb">\\fileserver\...\Pedidos tiendas\pedidos.mdb</a>
<b>Actualizar Pendientes Servir</b>	Actualitza les dades dels aplicatius utilitzats per a controlar les comandes pendents de servir al client final. Així es pot disposar de la informació actualitzada pel que fa a la disponibilitat dels articles a servir al client, en aplicatius posteriors. <a href="\\fileserver\...\pendientes servir y3\Actualizar pdtes servir.mdb">\\fileserver\...\pendientes servir y3\Actualizar pdtes servir.mdb</a>
 <b>Borrar Etiquetas</b>	<b>Esborra</b> uns documents que es generen en el sistema ERP diàriament per la impressió d'etiquetes d'articles utilitzant una sèrie de documents temporal. <a href="\\Server\C\informes josema\borrar etiquetas.mdb">\\Server\C\informes josema\borrar etiquetas.mdb</a>
<b>Pedidos Tiendas</b>	Llistats per a poder llistar el material que rebran les diferents botigues, des de altres botigues o el magatzem central, i així poder saber a qui va destinat cada article que els hi arribi amb la remesa del transportista. <a href="\\fileserver\...\Pedidos tiendas\x.mdb (donde x = nombre de la tienda)">\\fileserver\...\Pedidos tiendas\x.mdb (donde x = nombre de la tienda)</a>
<b>Pendientes Recibir Front</b>	Permet llistar els traspasos pendents d'acceptar dins el programari "Front" (TPV), des de les diferents botigues. És important acceptar-los, ja que sinó aquests articles del traspàs, no entrarien dins l'estoc del seu magatzem. <a href="\\Fileserver\userdata\Access database\pdtes recibir front\pdtes.mdb">\\Fileserver\userdata\Access database\pdtes recibir front\pdtes.mdb</a>
<b>Pendientes de Servir Tiendas</b>	Gestió de les comandes pendents de servir al client final des de les botigues (recollida a la botiga). Des d'aquest aplicatius poden visualitzar l'estat de les comandes pendents, així com anotar comentaris i notes en el seguiment de la comanda (disponibilitat, contacte amb el client, ...). <a href="\\Fileserver\userdata\Access database\pendientes servir y3\pendientes.mdb">\\Fileserver\userdata\Access database\pendientes servir y3\pendientes.mdb</a>
<b>Pendientes Servir (Central / Web)</b>	Igual que l'anterior aplicació, però en aquest cas gestiona les comandes pendents de servir directament al client des del magatzem central (enviament a domicili). <a href="\\Fileserver\userdata\Access database\pendientes servir\pendientes.mdb">\\Fileserver\userdata\Access database\pendientes servir\pendientes.mdb</a>
<b>Servir Todo Tiendas</b>	Utilitzat per a servir el material de comandes pendents que s'envien directament al client final. L'aplicatiu té dos funcionalitats bàsiques: <ol style="list-style-type: none"> <li>1. Moure el material des de les botigues cap al magatzem central, si no hi ha estoc disponible en el central.</li> <li>2. Una vegada rebut el material, generar els albarans i preparar el material per enviar-lo al client final.</li> </ol> <a href="\\Fileserver\userdata\Access database\servir desde tiendas\servir TODO tiendas.mdb">\\Fileserver\userdata\Access database\servir desde tiendas\servir TODO tiendas.mdb</a>
<b>Quitar de traspasos</b>	Extreu dels traspasos d'articles des de la central cap a les botigues, que es preparen durant el cap de setmana, articles que poden haver passat a ser necessaris per alguna comanda pendent d'un client o botiga durant l'impàs de temps del cap de setmana. <a href="\\Fileserver\userdata\Access database\pedidos-traspasos\traspasos.mdb">\\Fileserver\userdata\Access database\pedidos-traspasos\traspasos.mdb</a>

<b>Caja Almacén</b>	Llistat de caixes utilitzat per a organitzar el material a preparar pels enviaments a domicili dels clients finals. <a href="\\Fileserver\userdata\Access_database\cajas_almacen\cajas_almacen.mdb">\\Fileserver\userdata\Access_database\cajas_almacen\cajas_almacen.mdb</a>
<b>Etiquetas Almacén</b>	Genera un llistat d'etiquetes per a les caixes, una vegada hem especificat certs valors d'entrada (botiga, sèrie, nº inicial i nº final). <a href="\\Fileserver\userdata\Access_database\EtiquetasAlmacen\Etiquetas_almacen.mdb">\\Fileserver\userdata\Access_database\EtiquetasAlmacen\Etiquetas_almacen.mdb</a>

### 5.1.2 Bloc 2: Articles, estocs i inventaris

Nom	Breu Descripció
<b>Artículos</b>	Permet crear articles directament en la base de dades del ERP, aprofitant la creació per poder generar les talles i colors de l'article, així com els preus en diferents tarifes i estocs mínims. <a href="\\Fileserver\userdata\Access_database\articulos\articulos.mdb">\\Fileserver\userdata\Access_database\articulos\articulos.mdb</a>
<b>Mínimos</b>	Aplicació per a actualitzar els valors mínims d'estoc desitjats per diferents articles i diferents magatzems especificats en un full de càlcul d'entrada en format <b>Microsoft Excel</b> . <a href="\\proliant\c\informes_josema\minimos.mdb">\\proliant\c\informes_josema\minimos.mdb</a>
<b>Regularizaciones</b>	Permet llistar, segons diferents criteris i paràmetres, regularitzacions efectuades en diferents botigues o a nivell global, per unes dates concretes o per mesos. <a href="\\Fileserver\userdata\Access_database\regularizaciones\regularizaciones.mdb">\\Fileserver\userdata\Access_database\regularizaciones\regularizaciones.mdb</a>
<b>Artículos en Tienda</b>	Aplicació utilitzada en els botigues per a la visualització dels estocs de certs articles en la seva pròpia botiga o en les altres botigues de l'empresa. <a href="\\Fileserver\userdata\Access_database\stock_tienda\articulos_en_tienda">\\Fileserver\userdata\Access_database\stock_tienda\articulos_en_tienda</a>
<b>Poner Ubicaciones</b>	<b>Actualitza</b> la ubicació dels articles del magatzem central, a partir d'un full de càlcul en format <b>Microsoft Excel</b> . Per fer-ho es basa en la lectura efectuada amb un terminal de ma, de les diferents ubicacions del magatzem i els articles que hi ha guardats, llegint el codi de barres identificador, tant de les ubicacions com els articles. <a href="\\proliant\c\informes_josema\poner_ubicaciones.mdb">\\proliant\c\informes_josema\poner_ubicaciones.mdb</a>
<b>Actualizar Peso</b>	Permet modificar els diferents articles de la base de dades del ERP, perquè actualitzi el camp associat al pes de l'article. Per fer-ho es basa en un barem emmagatzemat en una taula local, on associa diferents pesos a diferents famílies, subfamílies, departaments i seccions. <a href="\\proliant\c\pesos\actualizar_peso.mdb">\\proliant\c\pesos\actualizar_peso.mdb</a>

### 5.1.3 Bloc 3: Tarifes i preus

Nom	Breu Descripció
<b>Actualizar Tarifas España</b>	<p>Procés automàtic que s'encarrega de calcular i regenerar certes tarifes de preus associades als articles. Partint d'unes tarifes bàsiques o "mare", regenera la resta de tarifes aplicant els criteris que pertoca a cada tarifa.</p> <p><a href="\\Server\C\informes josema\actualizar tarifas esp.mdb">\\Server\C\informes josema\actualizar tarifas esp.mdb</a></p>
<b>Recuperar PVP</b> 	<p>Guarda una còpia de una de les tarifes bàsiques del sistema ERP principal, en una taula local de la pròpia aplicació. En cas d'algun error humà a l'eliminar la tarifa, sempre es pot recuperar més ràpidament que no pas accedint a les còpies de seguretat de les bases de dades SQL Server emmagatzemades en una cinta magnètica.</p> <p><a href="\\Server2\e\josema\recuperar_pvp.mdb">\\Server2\e\josema\recuperar_pvp.mdb</a></p>
<b>Importación Oferta Andorra</b>	<p>Utilitza la web com a sistema d'intercanvi de dades o emmagatzematge intermig per tal de importar els preus de la tarifa d'oferta de la base de dades de la seu d'Andorra dins la base de dades de la seu d'Espanya.</p> <p><a href="\\Server2\e\josema\importacion oferta andorra.mdb">\\Server2\e\josema\importacion oferta andorra.mdb</a></p>
<b>Exportación PVP España</b>	<p>Utilitza, com en el cas anterior, la base de dades de la web per exportar els preus de la tarifa PVP de la base de dades d'Espanya, per una posterior importació en la base de dades d'Andorra.</p> <p><a href="\\Server2\e\josema\exportacion pvp espanya.mdb">\\Server2\e\josema\exportacion pvp espanya.mdb</a></p>

### 5.1.4 Bloc 4: CRM, consultes i altres

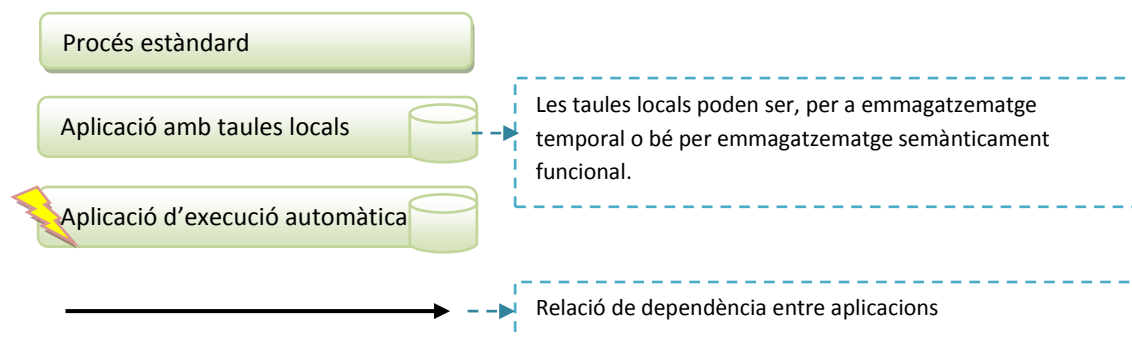
Nom	Breu Descripció
<b>Actualizar Bonos</b>	<p>Actualitza l'estat de les targetes regal perquè estiguin actualitzades tant en la base de dades de l'ERP com en la de la web per que no es puguin consumir més d'una vegada en les compres.</p> <p><a href="\\Server2\c\targetas regalo\Actualizar bonos.mdb">\\Server2\c\targetas regalo\Actualizar bonos.mdb</a></p>
<b>Motos Clientes</b>	<p>Permet assignar a un client, el model de moto del que disposa per, posteriorment, poder personalitzar l'oferta d'articles que s'ajustin a la seva moto. El client ens confirma ja en el moment de l'inventari, que aquesta aplicació estava en procés de creació i actualment es vol deixar de banda per fer-lo més acuradament en un futur.</p> <p><a href="\\Fileserver\userdata\Access database\motos clientes\motos clientes.mdb">\\Fileserver\userdata\Access database\motos clientes\motos clientes.mdb</a></p>
<b>Tarjetas Fidelización</b>	<p>Utilitzat per a actualitzar les dades de les targetes de fidelització de clients. Associa els nous clients creats a una targeta acumulativa de punts per compres utilitzant el codi intern de client del ERP.</p> <p><a href="\\Fileserver\userdata\Access database\fidelizacion\tarjetas fidelizacions.mdb">\\Fileserver\userdata\Access database\fidelizacion\tarjetas fidelizacions.mdb</a></p>
<b>Explotación</b>	<p>Permet generar llistats de les explotacions comptables de les diferents botigues, desglossant-les en base d'una prèvia parametrització.</p> <p><a href="\\Proliant\c\explotacion11\explotacion.mdb">\\Proliant\c\explotacion11\explotacion.mdb</a></p>
<b>Anticipos Clientes</b>	<p>Aplicació utilitzada per a actualitzar els noms dels clients en els pagaments avançats dels clients en els assentaments comptables.</p> <p><a href="\\Server2\ANTICIPOS CLIENTES.mdb">\\Server2\ANTICIPOS CLIENTES.mdb</a></p>

## 5.2 Relacions de dependència entre els diferents blocs d'aplicacions

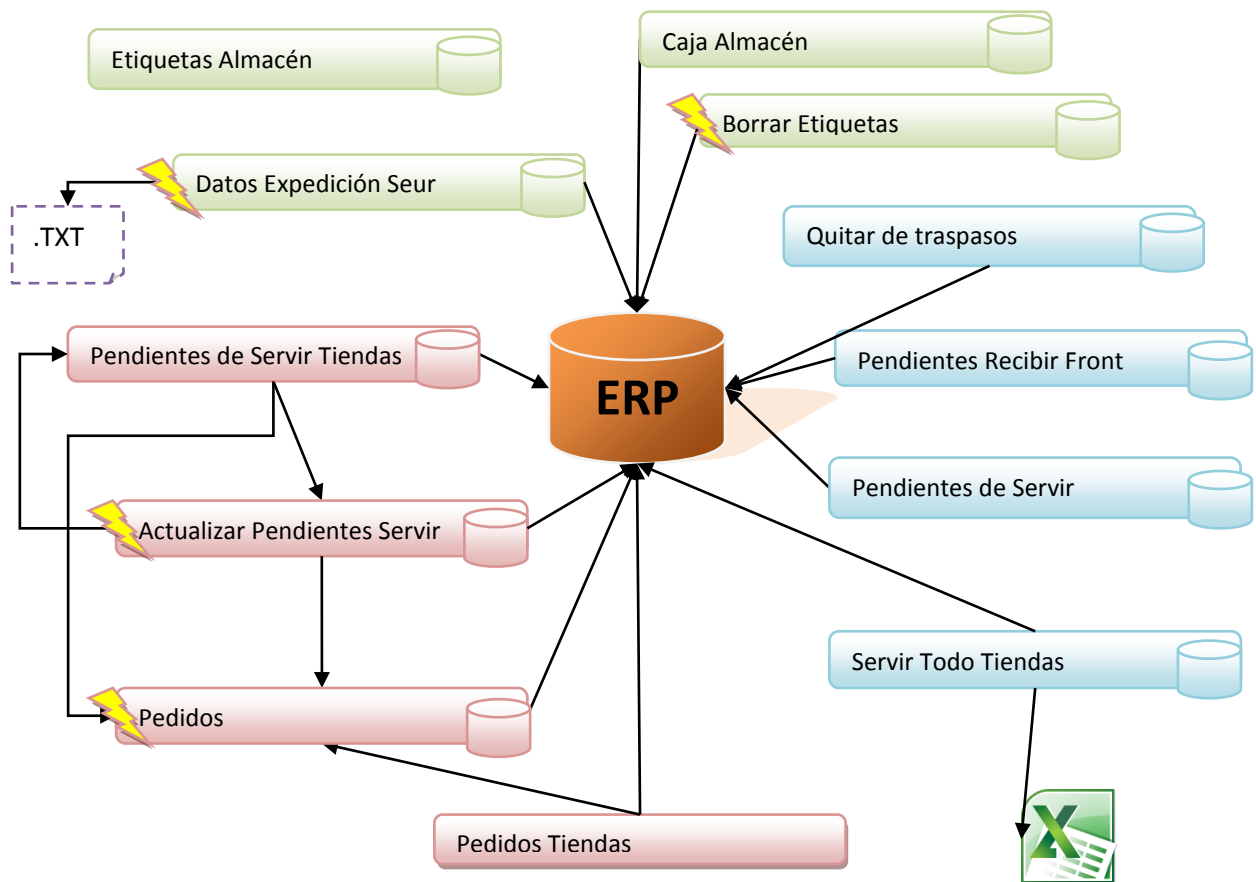
Una vegada inventariades totes les aplicacions del sistema satèl·lit, cal identificar les dependències actuals entre les aplicacions de diferents blocs o d'un mateix bloc. Com hem apuntat anteriorment, algunes de les taules d'aquestes aplicacions de *Microsoft Access* contenen enllaços no només cap a la base de dades principal, sinó a altres de les aplicacions satèl·lit.

En aquest apartat doncs, anotem mitjançant relacions gràfiques, les diferents dependències entre els diferents blocs d'aplicacions.

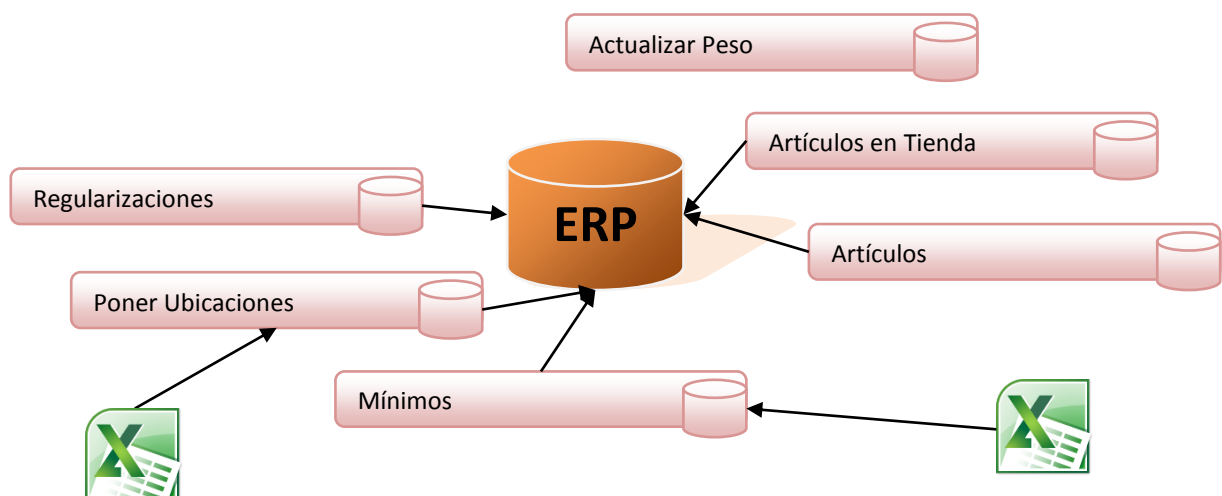
Per a poder interpretar correctament els gràfics de dependències, adjuntem a continuació una llegenda descriptiva dels diferents elements que intervenen en ells:



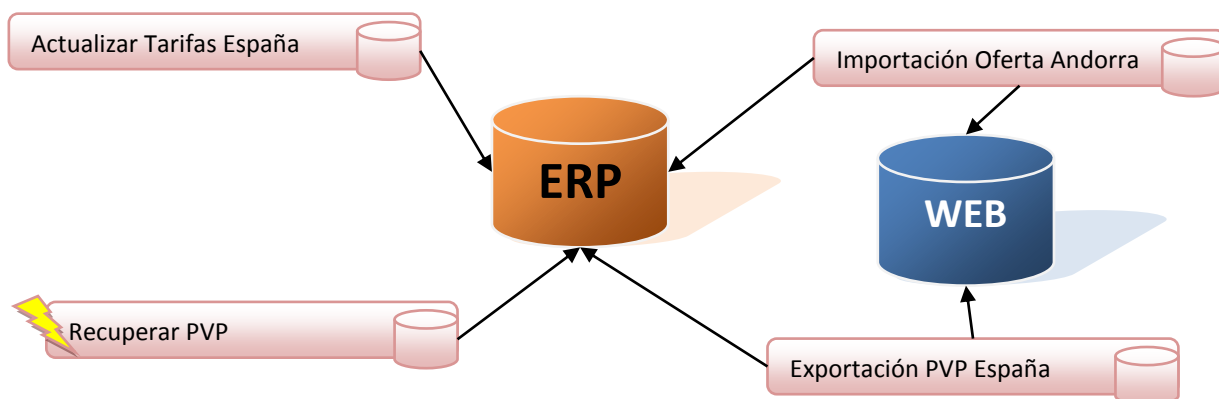
## 5.2.1 Bloc 1: Logística



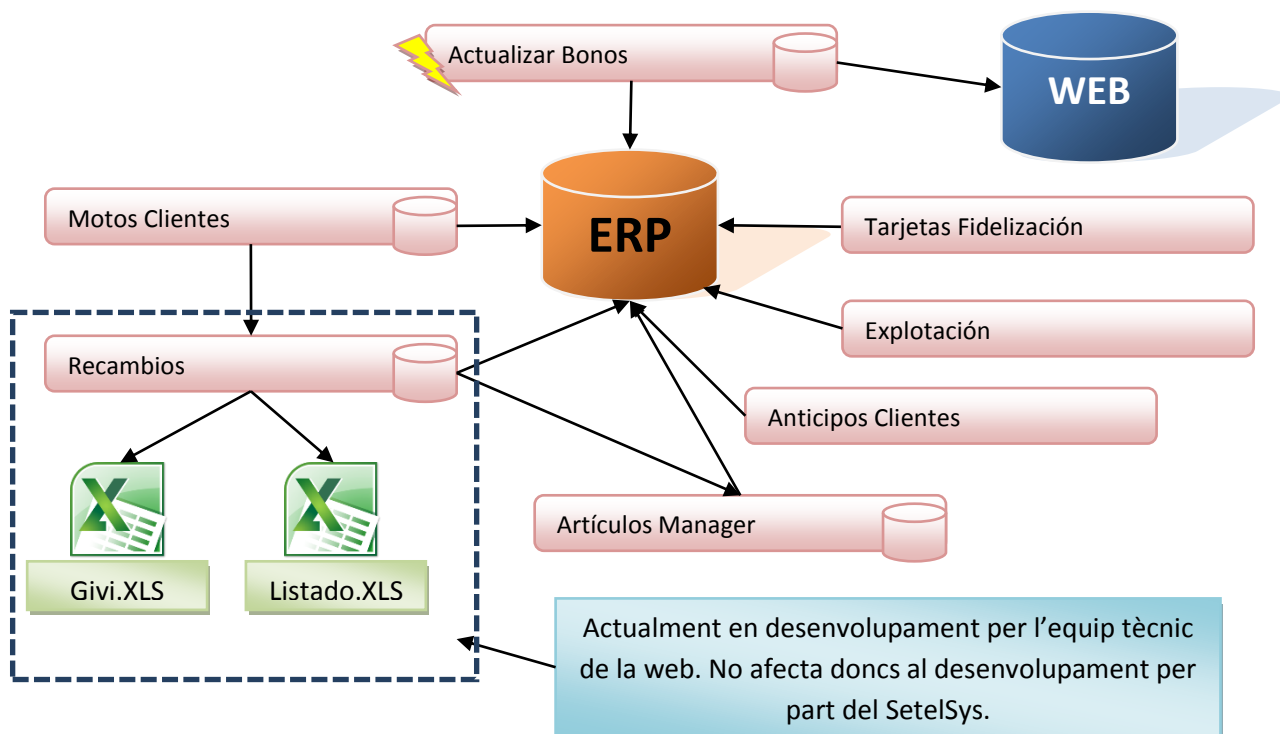
## 5.2.2 Bloc 2: Articles, estocs i inventaris



### 5.2.3 Bloc 3: Tarifes i preus



### 5.2.4 Bloc 4: CRM, consultes i altres



### 5.2.5 Conclusions sobre les dependències entre aplicacions

Com s'ha pogut veure en els gràfics dels apartats anteriors, el nivell de dependència de les aplicacions entre si, és exageradament elevat. Com hem anotat en anteriors punts, això implica:


- Efectes col·laterals no desitjats a l'efectuar canvis sobre una aplicació amb dependències associades.
- Si alguna aplicació en la cadena de dependències pateix algun error d'execució i les seves dades no resulten actualitzades correctament o incoherents, pot afectar la resta d'aplicacions amb les que existeix algun tipus de dependència.
- Es fa molt difícil establir un mecanisme de control en un marc de treball tant fragmentat, disseminat i sense mecanismes de control d'errors nadius.
- La distribució de totes les aplicacions en diferents plataformes hardware, fa que en cas de caiguda d'algun dels servidors, la resta d'aplicacions quedin inconnexes i indueixin a errades o incoherències de dades.
- Excessiva dependència entre aplicatius que fa difícil l'escalabilitat i modificabilitat del sistema, a part d'existir diferents punts on es pot trencar la coherència de les dades i els resultats.





### 5.3 Completesa i vigència de les aplicacions

Després d'haver recollit i catalogat les diferents aplicacions existents al sistema actual, i haver-ne estudiat la funcionalitat i l'objectiu, cal determinar amb el client quines d'aquestes aplicacions són completes pel que fa a la seva funcionalitat i vigents per la seva utilització.

En aquest punt ens reunim amb el client i analitzem el resultat del inventari per aconseguir fer un primer filtrat d'aplicacions on cal focalitzar l'anàlisi i especificació. Citarem a continuació els diferents blocs, detallant-ne les aplicacions que els componen tal i com hem vist en l'apartat 5.1 (*Inventari d'aplicacions*) i anotant si aquestes són completes funcionalment i vigents.

*Nota: Recordem que, com en l'anterior apartat, per anotar que l'aplicatiu és un procés automatitzat, utilitzarem la icona:* 

#### 5.3.1 Bloc 1: Logística

Nom		Completa	Vigent
Datos Expedición Seur		✓	✓
Pedidos 		✗	✓
Actualizar Pendientes Servir		✗	✓
Borrar Etiquetas 		✓	✓
Pedidos Tiendas		✗	✓
Pendientes Recibir Front		✓	✓
Pendientes de Servir Tiendas		✗	✓
Pendientes de Servir (Central/Web)		✗	✓
Servir Todo Tiendas		✗	✓
Quitar de traspasos		✓	✓
Caja Almacén		✗	✓
Etiquetas Almacén		✓	✓

### 5.3.2 Bloc 2: Articles, estocs i inventaris

Nom	Completa	Vigent
Artículos	✓	✓
Mínimos	✗	✓
Regularizaciones	⊘	✗
Artículos en Tienda	⊘	✗
Poner Ubicaciones	✗	✓
Actualizar Peso	✗	✓

### 5.3.3 Bloc 3: Tarifas i preus

Nom	Completa	Vigent
Actualizar Tarifas España	✗	✓
Recuperar PVP ⚡	✗	✓
Importación Oferta Andorra	⊘	✗
Exportación PVP España	⊘	✗

### 5.3.4 Bloc 4: CRM, consultes i altres

Nom	Completa	Vigent
Actualizar Bonos	✓	✓
Motos Clientes	⊘	✗
Tarjetas Fidelización	✓	✓
Explotación	⊘	✗
Anticipos Clientes	✓	✓

### 5.3.5 Conclusions sobre la completeness i vigència de les aplicacions actuals

Com hem pogut veure en els apartats anteriors, el sistema actual compren moltes aplicacions que tenen necessitats funcionals a satisfer, i algunes d'elles que ja no tenen cabuda en el sistema al no ser utilitzades.

Els motius pels quals algunes d'aquestes aplicacions han quedat obsoletes són:

- El sistema ERP Principal ha donat solució, en versions més actuals, a la capacitat de gestionar des del seu motor, algunes de les necessitats.
- Algunes aplicacions del sistema satèl·lit han absorbit i implementat part de les funcionalitats necessàries d'altres aplicacions.
- Canvis en la lògica de processos de l'empresa, que ha anul·lat la necessitat de certes aplicacions.

D'altra banda tota les noves necessitats detectades en les diferents aplicacions i que fa no considerar-les completes provenen dels requeriments de:

- Adaptar el sistema per donar resposta a les necessitats creixents de la lògica de processos de l'empresa. Cal donar una resposta al present i futur de l'empresa.
- Millorar la qualitat de les aplicacions en termes d'agilitat, facilitat, rapidesa i autonomia.

## 5.4 Funcionalitat actual de les aplicacions vigents

En molts casos, les aplicacions actuals del sistema satèl·lit efectuen unes consideracions, parametritzacions i criteris de decisió, de les quals el propi client desconeix el detall exacte. En part degut per una documentació inexistent dels processos i d'altra banda per tractar-se d'aplicacions molt vives i dinàmiques a nivell d'adaptació a l'entorn de l'empresa.

Per això és indispensable determinar el funcionament exacte de les aplicacions actuals per a poder-ho contrastar en el client tal i com veurem en el següent punt.

De cara a fer la memòria més amena i lleugera en la lectura, només introduïrem aquí l'anàlisi de les aplicacions que considerem més important i interessants de les existents, i que requereixen modificacions i/o ampliacions en les funcionalitats.

### 5.4.1 Aplicació "Pedidos" (Bloc 1: Logística)

#### Objecte de procés

Les comandes que ha de tractar aquest procés automatitzat, són totes aquelles les quals el número de sèrie de la comanda del client final compleixi la norma:

#### **E(1X)P**

On **X** és el número identificador de la sèrie de documents d'una botiga: E11P, E12P, E12P...

#### Descripció de l'aplicació

Procés automatitzat que juntament amb els aplicatius següents conforma el nucli neuràlgic del bloc logístic i del control de comandes pendents:

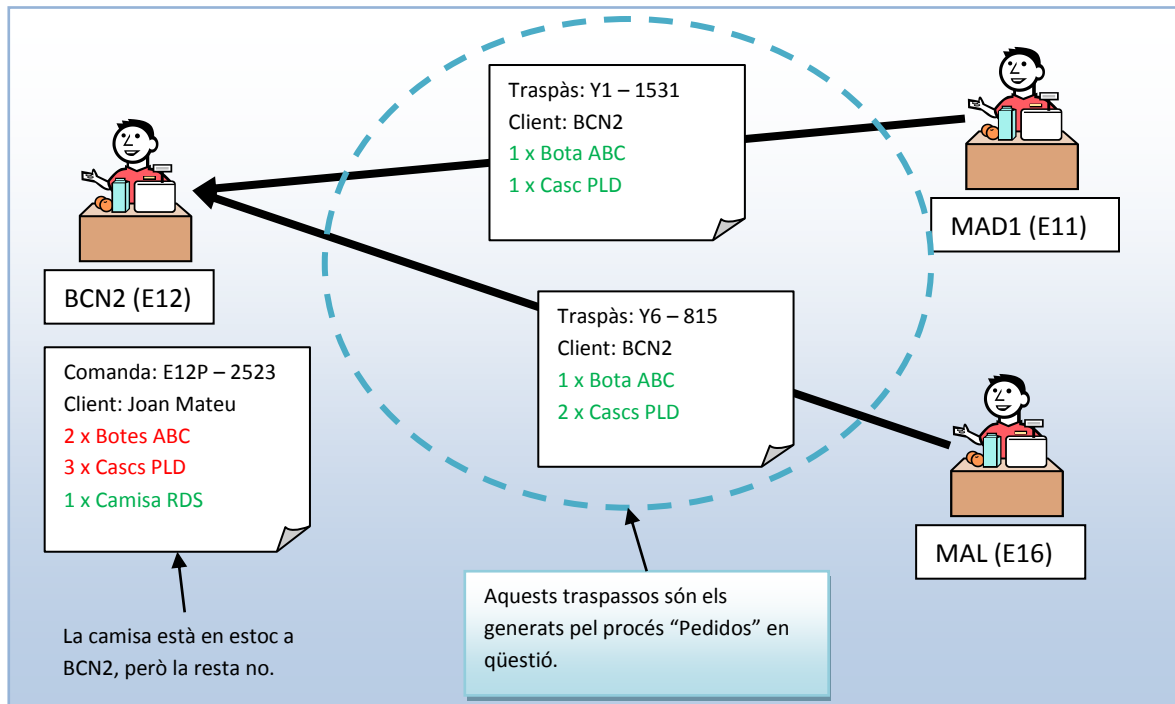
- Servir todo tiendas
- Actualizar pendientes de servir
- Pendientes de servir
- Pendientes de servir tiendas

El procés s'encarrega de generar els traspassos entre botigues per a subministrar els articles pendents per a comandes de clients que recolliran el material a les botigues. Recull totes les comandes pendents de servir al client, de la última setmana (últims 7 dies), i cerca si existeix alguna botiga amb existències d'aquests articles.

Si es el cas, genera un traspàs entre les botigues (sèrie documental Y(x) on x = botiga des d'on es traspassa el material). Aquest traspàs l'accepta la botiga que rep el material a través de l'aplicatiu de TPV i així queda en estoc al seu magatzem.

Posem un exemple per a visualitzar millor el seu funcionament. Cal considerar per aquest exemple:

- BCN2, MAD1, MAL: Són en aquest cas algunes de les botigues existents a l'empresa situades en diferents localitzacions geogràfiques (Barcelona, Madrid, Màlaga).
- Els traspassos són albarans de compra-venta dins l'empresa que es generen en l'ERP utilitzant una sèrie i numeració concreta per aquesta tipologia de documents.



Analitzem una mica el contingut del gràfic:

- En Joan Mateu visita la botiga de BCN2 i fa la seva comanda. Només un dels articles està en estoc, la resta queda pendent d'entregar al client.
- El dijous pel matí, a l'executar-se el procés "PEDIDOS":
  1. Es cerquen botigues on hi hagi estoc del material pendent per la comanda del Joan. Si en una botiga concreta existeix disponibilitat de tot el material, s'envia tot aquest material des d'una sola botiga. En cas contrari s'envia des de diferents botigues per tal de poder reservar el màxim de material per la comanda. En el cas del Joan, una part surt de Madrid i una altra de Màlaga.
  2. Es generen uns traspassos entre les botigues (un amb sèrie Y6, i un amb Y1) amb cost 0, per no desvirtuar les estadístiques de ventes ni el cost de l'article.
  3. Durant el matí del dijous, les botigues reben un llistat agrupat de tot el material que han d'enviar a altres botigues per satisfer comandes pendents. En el cas de Madrid, haurà de preparar "1 x Bota ABC" + "1 Casc PLD", a part de la resta de material necessari per altres comandes acumulades (aquí només analitzem el cas del Joan Mateu).

4. Les botigues s'envien el material entre elles durant la última hora del Dijous al matí.
5. El divendres, les altres botigues reben el material i accepten el traspàs rebut des del TPV del ERP, deixant en estoc tot el material. Al rebre el material, preparen el material i truquen al client, si és el cas. En aquest punt disposen d'un llistat de les comandes que es poden entregar completament i les que només es poden parcialment, per poder avisar al client sobre l'estat de la seva comanda. En el cas del Joan Mateu, ha rebut tot el material i se'l trucarà per comunicar-li.

### **Mancança d'estoc global**

En el cas de que no existeixi estoc en cap magatzem de les botigues existents en l'empresa (incloent el magatzem central), l'aplicatiu procedeix de la següent forma:

- Genera una comanda amb sèrie Y3 i magatzem E13 (central), amb el client associat a la botiga on està pendent la comanda.

Des del magatzem, es llista el material pendent de servir a les botigues i, si és el cas, es procedeix a fer les comandes de compra als proveïdors dins el ERP.

Una vegada arriba el material pendent des del proveïdor, s'envia el material a les botigues perquè puguin servir-lo al client.

### **Prioritats d'enviament entre botigues**

Existeix una prioritat d'assignació entre els magatzems quan existeix estoc en més d'un d'ells. Aquestes prioritats s'utilitzen per diferents criteris:

- a) Magatzems secundaris: Sovint interessa treure material primerament de magatzems més secundaris i que per tant no requereixen de tant material en estoc. En aquest cas es posaran pròxims en la cadena de prioritats.
- b) Proximitat geogràfica: Per qüestions de costos de transport es vol minimitzar el cost del moviment de material entre botigues i per tant les botigues més pròximes geogràficament requeriran estar al inici de la cadena de prioritats.

En tots els casos, el magatzem més prioritari des d'on s'ha d'intentar fer sortir el material és el magatzem central, per evitar treure estocs de les botigues, i les menys prioritàries són les botigues d'Andorra, per evitar costos aranzelaris.

Aquesta assignació de prioritats, actualment no és modificable d'una forma fàcil i àgil. Implica moltes modificacions de codi i estructura del mateix.

## Notificació dels llistats

Al finalitzar tot el procés, l'aplicació envia un missatge a través de correu electrònic a cada una de les botigues amb un informe que conté tota la llista de clients ordenats alfabèticament, els quals tenen reserves fetes a la botiga durant les dates compreses pel tractament de l'aplicació. Així tenen un resum del material que els hi arriba de les diferents botigues.

Els camps que conté aquest informe son:

Nom Client	Sèrie	Número	Telèfon Client	Articles			Magatzem
				Referència	Color	Talla	

## Llistats de control

Associada a aquesta aplicació, es requereix d'uns llistats de control per a cada una de les botigues per a permetre un correcte seguiment de les comandes:

1. Llistat de material reservat, pendent de recollir pel client final. Els camps que haurà de contenir són:

Client	Comanda (Núm.)	Article				Informació de l'estoc				Data Última Compra
		Ref.	Color	Talla	Desc.	Unit. Reservades	Unit. Estoc	Estoc Total	Quantitat Reserves	

2. Llistat del material pendent de rebre des de la central (de la sèrie Y3)

## Faltes d'estoc

En certs casos es poden produir desajustaments entre l'estoc que indica el sistema i l'estoc realment existent o disponible a la botiga (algunes vegades errors inherents a l'ERP, en d'altres errors d'origen humà).

En aquest cas utilitzen una tècnica una mica precària, basada en insereix una referència especialitzada per aquests casos anomenada "DOBLAR". Això fa que el mateix sistema actual detecti aquestes línies i les interpreti com que l'article de la línia anterior, tot i que en el sistema ERP disposa d'estoc, des de la botiga han detectat que realment aquest estoc no existeix.

Si no utilitzen aquest mecanisme, el sistema actual creu que realment ja disposa d'aquest article a la pròpia botiga i per tant no li fa arribar el material d'altres on també n'hi hagi.

Exemplifiquem com l'usuari entra una comanda E1(x)P, utilitzant la referència "DOBLAR" per tal de que el casc de la línia anterior no es consideri en estoc dins la seva botiga (per què l'usuari ha detectat que l'estoc del sistema i el real no quadren):

Referència	...	...	Unitats	...
C.ARA.DSC	...	...	1	...
DOBLAR	...	...	1	...

## Programació del procés

Hora	Periodicitat
06:30	Dijous de cada setmana



### 5.4.2 Aplicació “Servir Todo Tiendas” (Bloc 1 – Logística)

#### Objecte de procés

De forma resumida, aquesta aplicació tracta totes les comandes que s’han generat:

- A través de la pàgina web comercial de l’empresa.
- A través de la línia d’atenció telefònica.
- A través de la botiga, però que el client desitja que el material li sigui enviat directament al seu domicili (quan hi hagi disponibilitat).

En concret, les comandes que ha de tractar aquesta aplicació, són totes aquelles les quals:

**( Sèrie = E(1X)P ) ^ ( Magatzem = E13 )**

On **E1XP** és el número identificador de la sèrie de documents associats a la central: E13P, E15P.

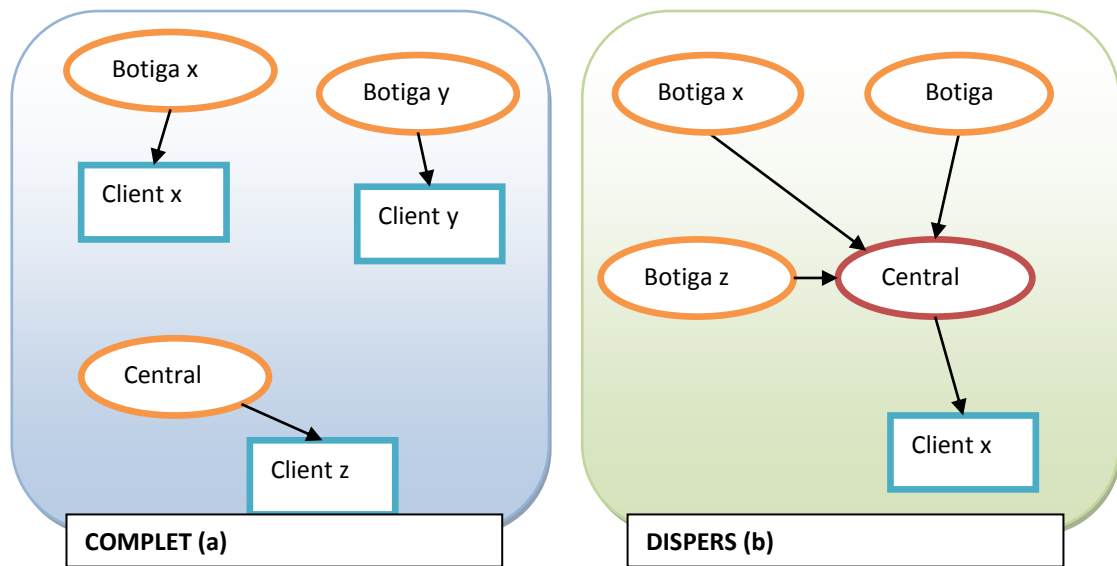
On **E13** és el magatzem central.

#### Descripció del procés

Durant tots els dies de la setmana, des del magatzem central, es processa i controla totes les comandes que s’han d’enviar directament al client final. Hi ha dos possibles escenaris quan parlem de servir material directament al client:

- a) **Complet:** Totes les comandes que tenen tot el material en alguna de les botigues o dins el magatzem central, s’envia directament cap al client des del punt de disponibilitat completa.
- b) **Dispers:** Si una comanda disposa d’estoc per servir-la però l’estoc està distribuït en diferents magatzems de botigues i/o el central, aleshores es fa enviar el material des de les botigues que en disposen cap al magatzem central, on una vegada rebut el material s’enviarà directament al client.

Anotem a continuació uns diagrames conceptuals d'aquests dos escenaris:



### Periodicitat

Aquests procediments en la gestió de l'enviament de comandes al client final, s'utilitzen amb la següent planificació temporal:

- a) Diàriament en el cas del COMPLET
- b) Dilluns, dimecres i divendres en el cas del DISPERS.

### Criteris i parametrització

Actualment els criteris en la programació d'aquesta aplicació estan integrats en el procés de codificació, així com tota possible parametrització. És per aquest motiu que farà falta redissenyar al complet aquests criteris i parametritzacions. Tot i que en l'apartat d'ampliacions de funcionalitats ho anotarem detalladament, us donem un breu resum d'alguns dels criteris que caldrà redissenyar:

- Preferències d'enviament de material per ubicació geogràfica.
- Balanceig d'estocs
- Preferències de magatzems
- Disponibilitat dels magatzems
- ...

**Funcionament intern del procediment DISPERS:**

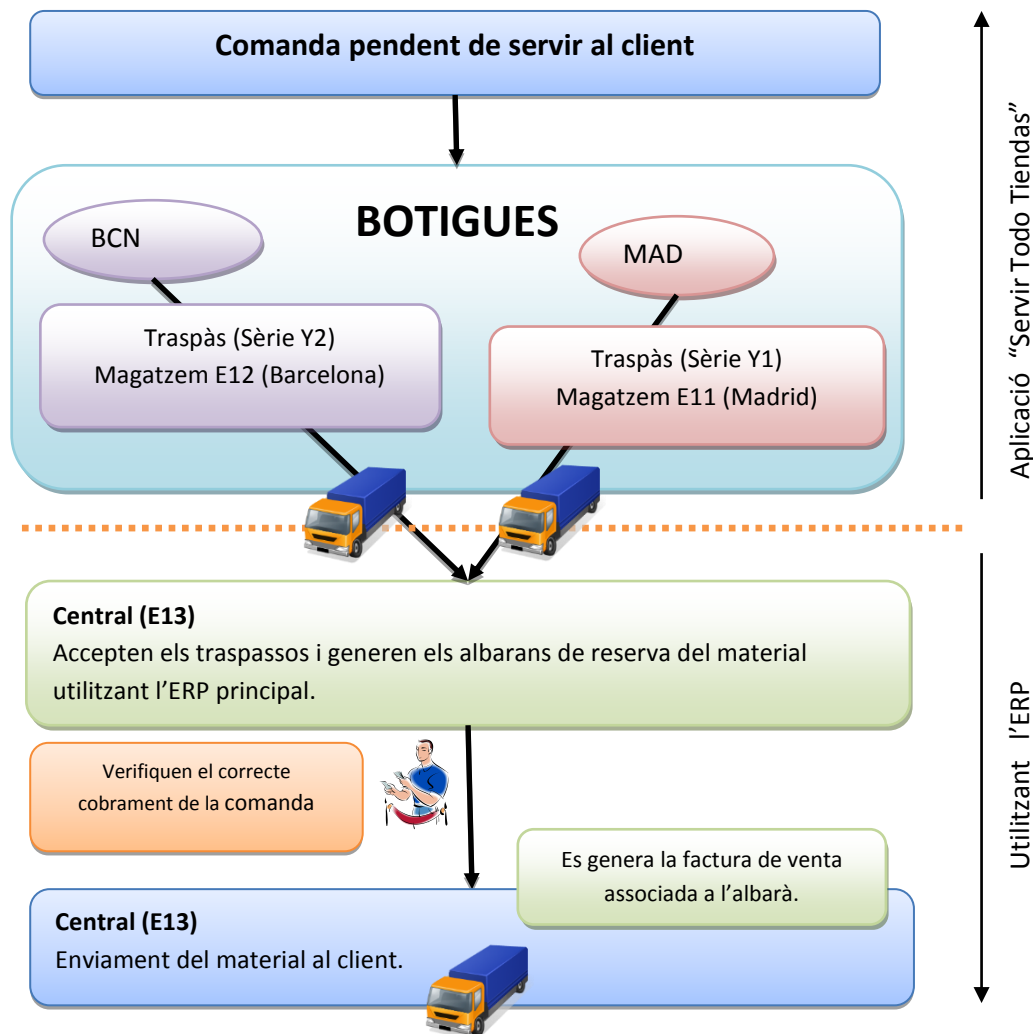
És important identificar que aquest procediment s'executa una part a través de l'aplicació "Servir Todo Tiendas" en qüestió mentre que la resta s'executa utilitzant les eines disponibles de l'ERP:

A través de l'aplicació "Servir Todo Tiendas"

1. Quan el procés revisa les comandes que només es poden servir dispersament, genera albarans de traspàs del material de les diferents botigues per enviar cap a la central.
2. Les diferents botigues reben la notificació d'aquests traspassos de material que hauran d'enviar cap a la central.
3. Les botigues serveixen el traspàs o en cas de falta d'estoc avisen a la central de la falta de disponibilitat (degut a un error de l'estoc en l'ERP, estoc en demostració, defectuós, ...).
4. Es genera un llistat on apareix tot el material que rebran de cada botiga i a quina comanda de client va assignat.

A través de l'ERP:

5. A la central reben el material de les diferent botigues i accepten els traspassos perquè el material quedi computat dins el magatzem central i descomptat definitivament dels magatzems de les botigues d'on prové.
6. Es generen els albarans per reservar el material pel client utilitzant el sistema natiu integrat dins l'ERP.
7. Es reserva el material associat a les diferents comandes.
8. Es verifica l'estat correcte del pagament de les comandes reservades.
9. Es generen les factures de venda, les etiquetes d'enviament i s'envia el material al client.



### Funcionament intern del procediment COMPLET:

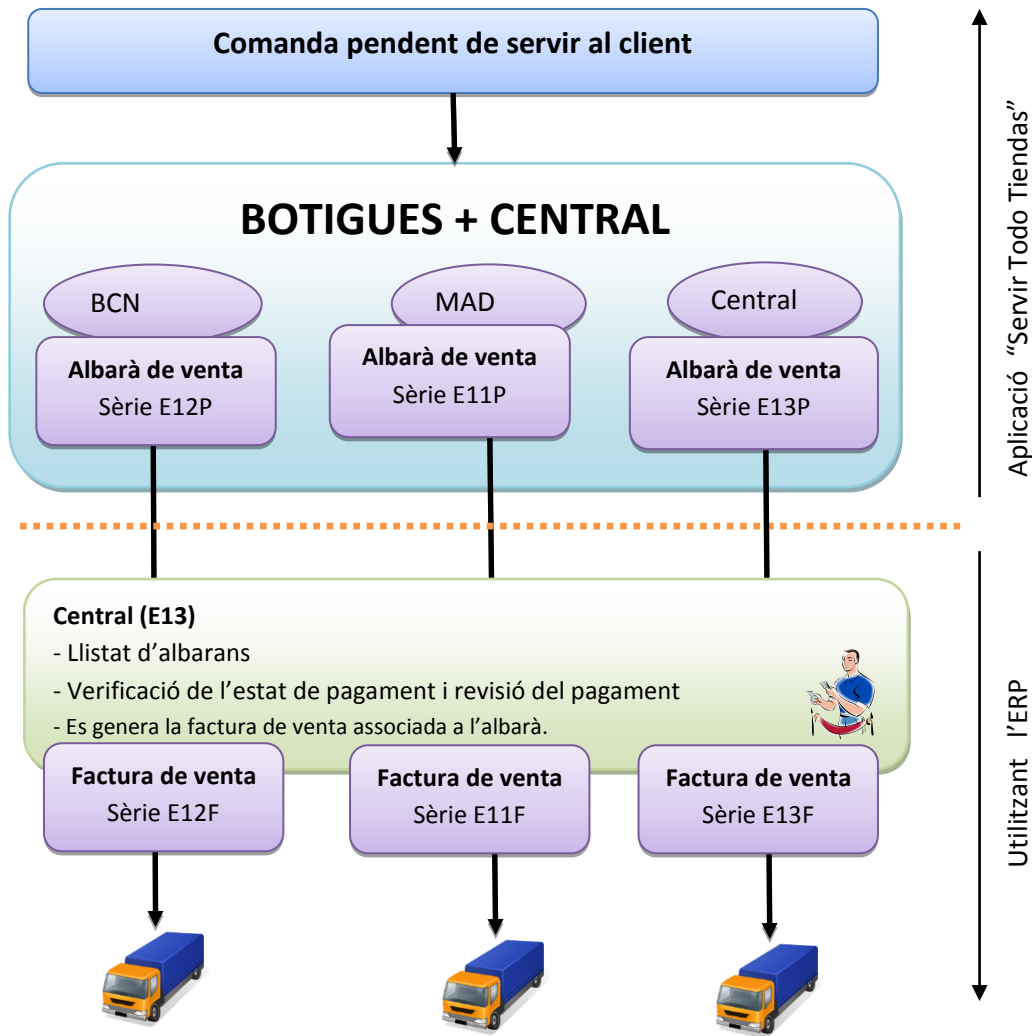
Com en el cas anterior, es genera una part a través de l'aplicació especialitzada, i una altra des del propi ERP.

A través de l'aplicació "Servir Todo Tiendas":

1. Es genera l'albarà de la comanda que té disponibilitat completa, ja sigui en el magatzem central o bé des d'una botiga.
2. Es genera un llistat dels albarans acabats de generar, desglossat per magatzem.

A través de l'ERP:

3. S'imprimeixen tots els albarans, un a un, en base del llistat anterior.
4. Se'n verifica el correcte estat del pagament.
5. Es generen les factures de venda, les etiquetes d'enviament i s'envia el material al client.



### Temporització cíclica dels processos

Per entendre tot el cicle complet d'aquest procés cal tractar-lo des de l'òptica de 2 dies consecutius (Dia 1, Dia 2), on veurem que i quan s'executa. Cada execució de l'aplicació "Servir Todo Tiendas" generarà un seguit d'albarans i traspassos que denotarem de la forma {AlbaransX}, {TraspassosX}.

L'hora indicada és aproximada i indicativa, ja que depèn una mica del volum de material a preparar, possibles incidències que sorgeixen, disponibilitat dels treballadors, ...

Hora	Dia 1	Dia 2
07:30	Verificació de l'estat del pagament. Si tot està correcte, generen una comanda dins de l'ERP.	
8:00	Execució Complet: {70Albarans1} Execució Dispers: {Traspassos1}	Execució Complet: {Albarans3}
08:30	Llistat de caixes per preparar des del magatzem central pel material de {Albarans1}	Llistat de caixes per preparar des del magatzem central pel material {Albarans3}
09:00	Les botigues i central preparen els albarans que els hi pertoqui de la llista {Albarans1}	Les botigues i central preparen els albarans que els hi pertoqui de la llista {Albarans3}
		La central prepara el material de la llista de caixes del dia anterior dels {Albarans2}
		S'envia el material dels {Albarans2} cap al client.
10:00	Les botigues preparen el material de {Traspassos1} per enviar cap a la central.	
13:00	Les botigues envien el material dels {Albarans1} cap al client i el material de {Traspassos1} cap a la central.	
15:00	La central accepta els traspassos {Traspassos1} de material provinent de les botigues.	
15:30	Generen els albarans de venda {Albarans2} pel material de {Traspassos1} des de l'ERP, tot i que el material no arribarà físicament fins el Dia 2.	
16:00	Llistat de caixes per reservar el material dels {Albarans2}	

Aquest cicle de vida de dos dies es reparteix de forma continua durant la setmana, seguint el patró següent:

Dilluns	Dimarts	Dimecres	Dijous	Divendres	Dissabte
Dia 1	Dia 2	Dia 1	Dia 2	Dia 1	Dia 2

### 5.4.3 Aplicació “Pendientes” i “Pendientes Y3” (Bloc 1 – Logística)

Existeixen dos aplicacions per a fer un control i gestió de les comandes pendents de servir al client, depenent de si aquestes comandes s’han d’enviar directament al client final o bé si són comandes que recolliran a alguna de les botigues existents.

Primer de tot, definim una mica més acuradament que ha d’entendre el sistema per a **comanda pendent**:

Una comanda pendent és tota aquella que conté alguna línia referent a un article, la qual té unitats pendents d’entregar al client. Per això cal anotar que en les taules on s’enregistren les línies de cada comanda existeixen tres tipus d’unitats:

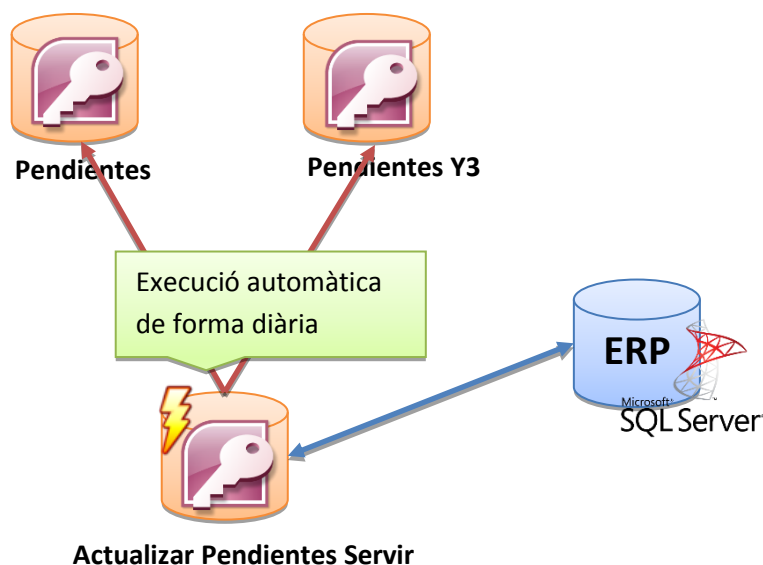
- **UNIDADESTOTAL** – N° total d’unitats que s’han de servir al client.
- **UNIDADESREC** – Unitats ja entregades al client (i per tant s’ha fet un albarà, tiquet o factura)
- **UNIDADESPEN** – Unitats pendents d’entregar al client (no s’ha fet albarà, ni tiquet ni factura)

*Per tant tota comanda, que tingui alguna de les línies associades amb les UNIDADESPEN>0, es considerarà una comanda pendent.*

Vist aquest aclariment tenim que, de les dos aplicacions:

- **“Pendientes”**: S’encarrega de visualitzar les comandes pendents d’enviar directament al client final.
- **“Pendientes Y3”**: S’encarrega de visualitzar les comandes pendents de recollir pel client a les diferents botigues, ja sigui perquè manca estoc o bé perquè el client encara no les ha recollit.

Aquests processos s’actualitzen de forma diària (a través d’un procés automàtic) que actualitza l’estat de les comandes pendents afegint les noves que es detectin, eliminant les que ja estan servides en la totalitat o actualitzant les que encara estan pendents.



### Funcionalitat d'ambdós aplicacions

Tot i que l'objectiu d'ambdós aplicacions explora comandes de naturalesa diferent, comparteixen les mateixes funcionalitats:

- Llistar les comandes i el material pendent en concret, així com les dades de contacte del client.
- Gestionar notes i comentaris a nivell intern pel seguiment, i informació de control sobre la comanda.
- Modificar la data prevista d'arribada del material i notificar canvis en aquesta data als usuaris.
- Notificar la necessitat de trucar al client (nacional o internacional).
- Llistar les comandes de proveïdor per l'article de la comanda pendent en qüestió, per obtenir informació extra sobre la previsió o estat de l'arribada de material.
- Unitats pendents de l'article de la comanda pendent, en altres comandes pendents d'altres clients.
- Cercar comandes pendents per un client en concret.
- Generar consultes directament al magatzem des de les botigues (sense existir una comanda associada) per conèixer la disponibilitat d'un article, present o futura.

També se'ns permet la generació de llistats per:

- Llistats de control per poder generar informes de les trucades pendents d'efectuar a clients i alarmes d'arribada de material (per verificar-ne l'exactitud i poder actuar amb antelació en la comunicació amb el client).
- Llistat de clients en espera, que tenen comandes pendents de servir, ja sigui per falta d'estoc o per que estem esperant la recollida del material a la botiga.
- Llistat de material que té associades faltes d'estoc.
- Llistat dels clients ja servits (es pot anar buidant aquesta taula històrica).



## 5.5 Contrastar les funcionalitats vigents amb el client

Després de l'etapa de recollida de funcionalitats vigents en el sistema, establim les corresponents reunions amb el client, entregant-li el corresponent informe fruit del punt 5.4, per tal de:

- Revisar meticulosament les diferents aplicacions
- Revisar-ne altre cop la completeness i vigència.
- Determinar noves necessitats en el sistema.
- Detectar necessitats actualment acomplertes però necessàries de revisió.

D'aquest punt en sorgeix el punt 5.6 que consisteix en el detall de les noves funcionalitats i rectificacions a realitzar en les aplicacions existents.

## 5.6 Ampliació de funcionalitats per part del client

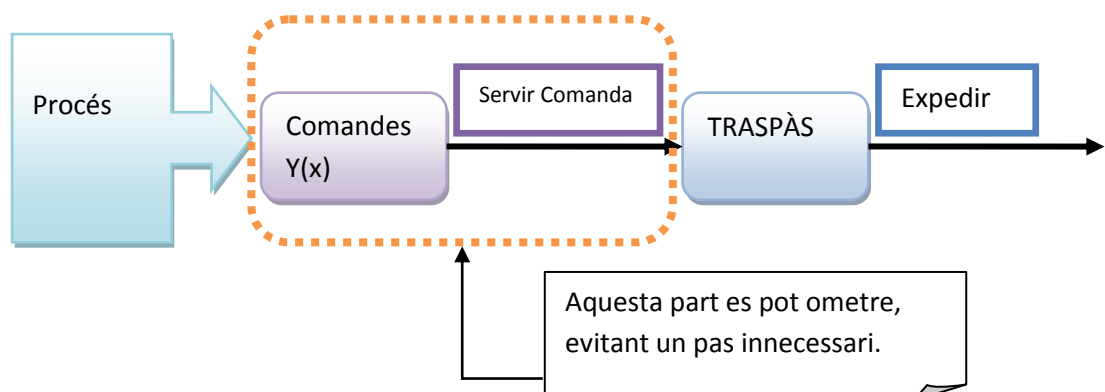
Com en l'apartat 5.4, ens centrem exclusivament en les aplicacions més important i interessants per l'anàlisi.

### 5.6.1 Aplicació "Pedidos" (Bloc 1 – Logística)

#### Canvi en la periodicitat i el flux de treball (*Workflow*) de les botigues i el magatzem central

El client vol modificar el paradigma de treball de l'aplicació per complir els següents requisits:

- No s'executi el procés un sol cop a la setmana, sinó dos cops al dia. Això permetrà que sempre es reservi el material per les botigues amb comandes pendents, el més aviat possible i evitar que comandes o compres posteriors consumeixin estoc que pertocaria a una comanda preferent per antiguitat.
- L'enviament de material es seguirà fent un cop a la setmana (minimitzar costos de transport), però la diferència és que durant la setmana s'hauran anat acumulant articles per reservar entre les botigues en diferents traspassos acumulatius. (Un traspàs acumulat per cada combinació de botiga origen – botiga destí). Això farà que l'estoc dels articles quedi reservat per les botigues que el demanen i per tant no es puguin servir a comandes posteriors.
- El llistat automàtic que actualment s'envia a les botigues, s'hauria d'enviar a l'encarregat en format de full de càlcul (*Microsoft Excel*). Per la resta de treballadors de la botiga hauria d'existir una interfície còmoda i senzilla per poder visualitzar en tot moment el material que han de reservar per les botigues i el que rebran durant aquesta setmana.
- Actualment aquesta aplicació genera comandes entre les botigues, cosa que implica que posteriorment han de transformar aquestes comandes generades a les botigues, en albarans de venta interns (traspassos). Es vol saltar aquest pas intermedi i generar directament els traspassos, evitant un pas extra a nivell de gestió per part de les botigues i que a la vegada sovint indueix a errors que podríem evitar.



- Cal agilitzar el procés de gestió de l'enviament del material associat a les comandes que les botigues fan al magatzem central, de forma que quan el material arriba al proveïdor, aquest mateix sistema sigui capaç de detectar aquestes comandes pendents i servir-les primer de tot (ja que són les comandes amb més antiguitat). Aquesta part també ha de formar part de l'execució continua de l'aplicació per assegurar que es minimitza el temps d'espera del client.

### **Objecte del procés**

- Actualment el sistema només tracta comandes pendents dins l'interval d'una setmana. Això provoca que comandes anteriors als 7 dies que utilitza en la cerca no es processin adequadament. L'aplicació ha d'analitzar en cada execució totes les comandes amb unitats pendents, ja que pot haver aparegut estoc en altres botigues (regularitzacions, devolucions, compres a proveïdors,...) o bé en el magatzem central (generalment compres a proveïdors).

### **Canvis en la gestió de faltes d'estoc**

Es vol eliminar l'ús de la tècnica del "DOBLAR" i que es puguin registrar de manera controlada les faltes d'estoc associades al magatzem d'una botiga o al central, perquè la resta d'aplicacions del sistema puguin tenir-les en compte en el càlcul de l'estoc real disponible dels articles. Aquest sistema de gestió de faltes haurà de permetre:

- Enregistrar faltes d'una manera àgil i còmoda des de les pantalles de control del material a reservar per les altres botigues o el magatzem central.
- Les dades que caldrà enregistrar en un falta són:
  1. Data i hora creació
  2. Usuari que la registra
  3. Motiu de la falta
  4. Data i hora de la solució
  5. Usuari que la soluciona
  6. Motiu de la solució.
- Poder llistar les faltes existents, i filtrar-les pel seu estat i magatzem d'origen.
- L'administrador o encarregat de les botigues, haurà de ser l'únic capaç d'eliminar faltes de forma manual. Els usuaris no administradors simplement podran entrar noves faltes en el sistema des de les pantalles corresponents o de forma manual des de la pantalla destinada per aquest objectiu.

### Prioritats i disponibilitats dels magatzems

- Cal dotar a l'aplicació de la capacitat de gestionar i modificar fàcilment les prioritats entre botigues pel que fa a la decisió dels moviments de material. Adjuntem a continuació la taula de prioritats desitjada i que caldria poder editar de forma àgil. Actualment no compleix aquesta estructura, però caldria que ho fes:

Magatzem d'on surt el material												
	E11	E12	E14	E15*	E16	E17	E18	E19	E1Z	E13	A11	A12
E11		7	6		4	3	5	2	8	1	9	10
E12	7		5		4	3	2	6	8	1	9	10
E14	8	7			3	2	4	5	6	1	9	10
E15	7	11	5		4	3	2	6	8	1	9	10
E16	8	7	3			2	4	5	6	1	9	10
E17	8	7	2		3		4	5	6	1	9	10
E18	8	7	4		3	2		5	6	1	9	10
E19	8	7	5		3	2	4		6	1	9	10
E1Z	8	7	6		3	2	4	5		1	9	10

(\*E15 – Magatzem virtual associat a la WEB)

- Cal permetre activar o desactivar temporalment el còmput d'algun dels magatzems de les botigues, en cas de festivitat local, per evitar que es demani material que arribaria amb demora. Pot ser que un article estigui present en diferents magatzems i es demani en un que no roman obert durant aquest dia. Segons el dia de la setmana en que es produeixi aquesta casuística, pot ampliar una setmana el temps d'entrega al client.

### Consideracions en el càlcul d'estoc disponible

L'aplicació ha de tenir en compte en tot moment, les unitats disponibles realment a la botiga. El sistema ERP principal ens informa de cada article i per cada magatzem de:

- Unitats pendents d'entregar (SE): Ens indica les unitats que hi ha pendents d'entregar al client final (ja sigui perquè no hi ha estoc o perquè estan pendents de recollir o entregar)
- Unitats en trànsit (ST): Ens indica la quantitat d'articles que estan en traspàs d'altres magatzems cap al nostre. Aquests traspàsos són referents a material enviat des de les botigues o la central cap al nostre magatzem.

- Unitats demanades a proveïdor (SP): Ens indica la quantitat d'articles demanats al proveïdor i pendents de rebre en estoc.
- Estoc físic (SF): Fa referència a l'estoc físic de l'article, sense tenir en compte si existeixen reserves, material que està en camí o material demanat al proveïdor.

Per tant les unitats realment disponibles, o estoc real (SR), que es poden enviar a altres botigues, han de respectar el càlcul de les diferents variables anotades i es pot expressar en la següent formula:

$$SR = SF + ST - SE$$

$$\text{<Estoc Real>} = \text{<Estoc Físic>} + \text{<Estoc en Transit>} - \text{<Estoc pendent d'entrega>}$$

**Nota:** El client no vol tenir en compte o considerar en aquest moment les unitats pendents de rebre del proveïdor, ja que això pot suposar retards en l'entrega de les comandes al client degut a la falta d'una data factible d'entrega del proveïdor a la pròpia empresa.

### Implicacions en altres aplicacions

Caldrà dotar a l'aplicació encarregada de la gestió, visualització i control de les comandes pendents, de forma que pugui disposar d'una consulta ràpida del seguiment de la comanda d'un client, per millorar la comunicació amb el client en casos d'anomalies i millorar l'anticipació a conflictes, incrementant el feedback entre el client i la botiga o la central.

### Gestió retroactiva

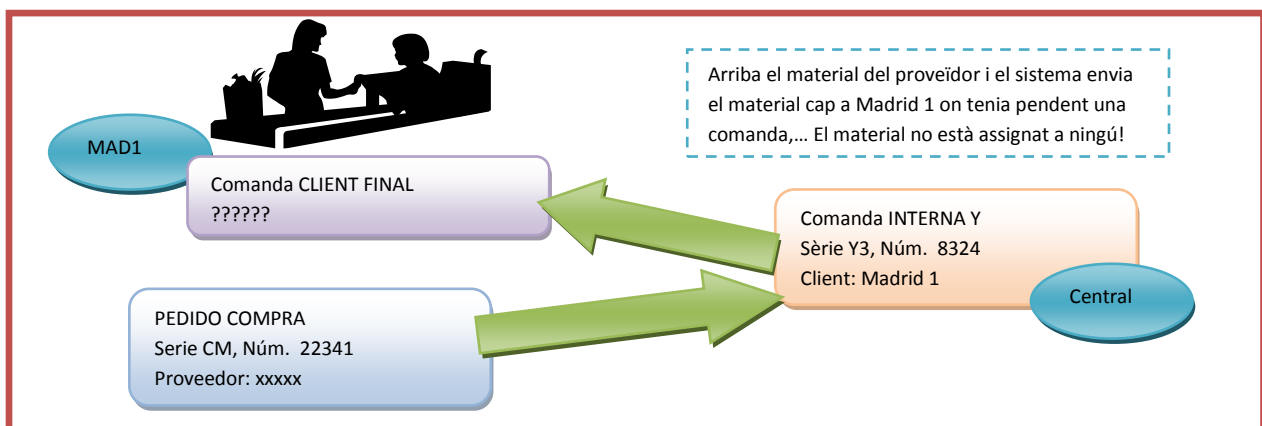
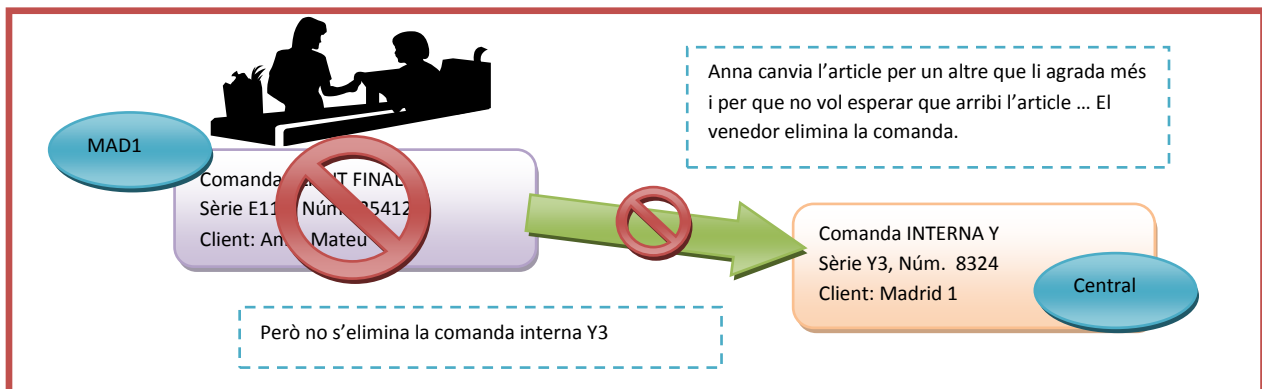
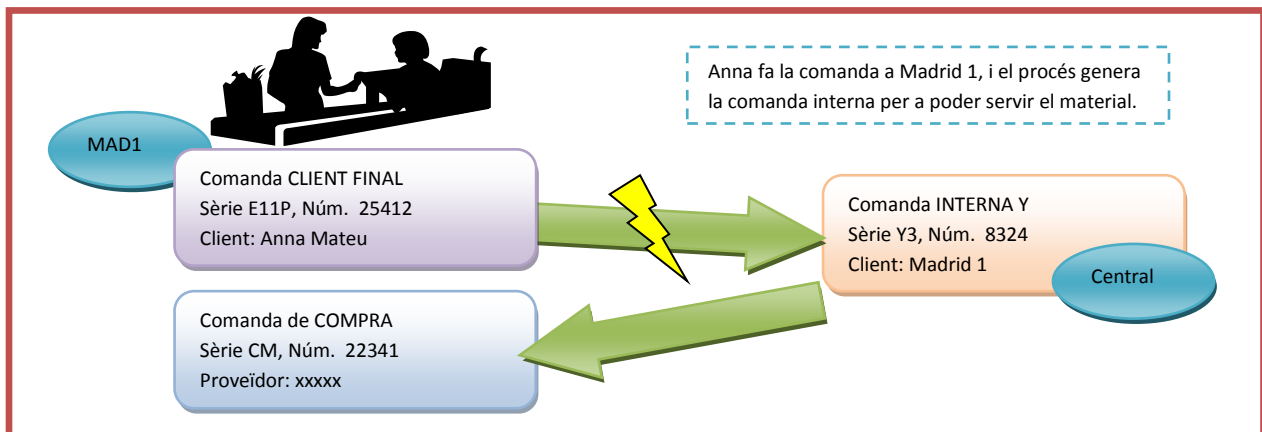
Sovint les botigues eliminen la comanda feta pel client final per diferents motius (el client no vol esperar més, vol canviar l'article, ...). En aquests casos queden descontrolades les comandes que en el seu moment s'havien generat a la central perquè s'encarregués de gestionar-ne la disponibilitat.

Quan arriba el material que s'ha demanat al proveïdor per aquesta comanda, s'acaba enviant a la botiga que l'havia demanat, sense necessitat real:

- Generant un sobre estoc innecessari.
- Generant uns costos de transport innecessaris.

El sistema hauria de ser capaç de controlar aquest escenari, i abans d'enviar el material associat a una comanda d'una botiga, verificar realment de la necessitat real d'estoc d'aquesta contrastant les unitats que té pendents, les que té en estoc real i les faltes que puguin existir.

Intentem exemplificar aquest escenari conflictiu mitjançant els gràfics següents:



### Altres funcionalitats desitjades

- Seria necessari afegir un sistema d'avís en línia, per recordar a les botigues els traspassos que tenen pendents d'acceptar i per tant d'incorporar en el seu estoc, ja que es detecta que en diverses ocasions es queden pendents en el sistema i provoquen problemes d'estocs al sistema.

### 5.6.2 “Servir Todo Tiendas” (Bloc 1 – Logística)

El client desitja una remodelació total d'aquesta aplicació per:

- Ajustar-lo a les necessitats reals de la lògica de negoci de l'empresa
- Evitar la dependència d'usuaris experts en el procés.
- Minimitzar el temps d'execució global.
- Fer-la més flexible i capaç de configurar-se amb uns paràmetres mínims.

#### Canvi en la periodicitat i el flux de treball (*Workflow*) de les botigues i el magatzem central

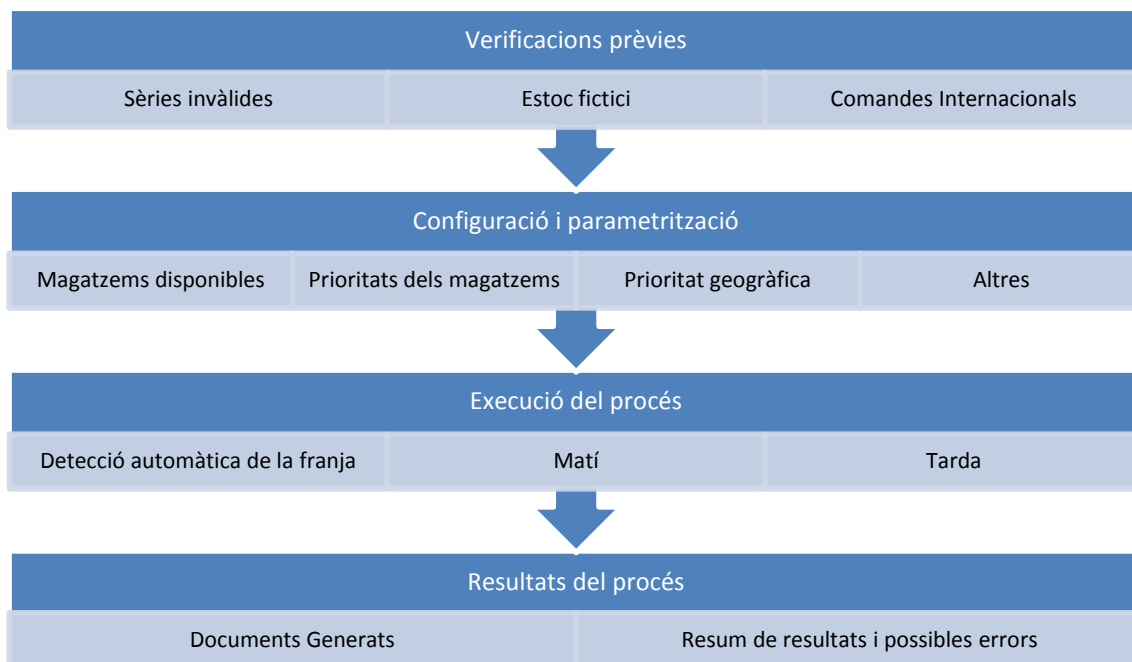
El client vol modificar el paradigma de treball de l'aplicació, de manera que compleixi els següents requisits:

- **Temporització:** No s'executi el procés en blocs de 2 dies (Dia1 + Dia2) sinó que s'executi de forma diària. En concret es vol plantejar dos esquemes especialitzats de funcionament, un pensat per l'execució durant el matí i un per la tarda, tot i que el sistema hauria de detectar-ho de forma automàtica en base de l'hora de l'execució.
- **Unificar el procés:** Actualment, i com hem vist en l'anàlisi de l'aplicació, el flux de treball està dividit en l'ús de dos aplicacions (ERP + Aplicació “Servir Todo Tiendas”). Es desitja que una sola aplicació s'encarregui del màxim possible de tasques dins del procés i unificar així la feina i reduint el temps d'execució i dependència manual. Hi ha un seguit de tasques que es poden automatitzar, però d'altres que hauran de seguir elaborant-se de forma manual.
- **Configuració:** Dotar a l'aplicació de la configuració necessària per poder parametritzar el procés en base d'uns criteris que analitzarem tot seguit.
- **Balanceig de l'estoc:** Es vol fer un ús intel·ligent del consum d'estoc disponible als diferents magatzems.
- **Automatitzar:** Actualment l'aplicació “Servir Todo Tiendas”, va acompanyat d'altres aplicacions que ajuden a finalitzar el procés (com per exemple el llistat de caixes) i altres procediments dins l'ERP (impressió dels albarans). Es vol automatitzar el màxim aquest procés de manera que s'executi tot en bloc i generi un resultat final amb poca interacció de l'usuari.
- **Generació de documents de suport:** Quan s'executi l'aplicació de “Servir Todo Tiendas” de forma automatitzada, es desitja que es generin els corresponents llistats de control i actuació per què les persones assignades a les diferents tasques puguin procedir amb la seva labor. Es voldrà assignar als diferents usuaris del sistema un seguit de documents que tenen permís per visualitzar i imprimir, de forma que cadascun d'ells només vegi els que li pertoca.

- **Control de sèries invàlides:** Degut a errors humans, de vegades s'entren comandes dins del sistema utilitzant sèries documentals inadequades. El sistema actual les tracta igualment generant resultats incoherents a nivell organitzatiu. Per tant es vol dotar al sistema d'un llistat de control de sèries invàlides abans d'iniciar l'execució del procés funcional.
- **Estoc fictici:** L'empresa utilitza un seguit d'articles per la venda a través de la pàgina web, que en realitat ja estan obsolets i sense estoc, però es poden assignar articles d'identiques prestacions sense cost extra pel client. Actualment aquests articles s'han de verificar manualment i per tant suposa una tasca força tediosa. Es vol integrar aquest sistema de control i canvi automàtic dels articles associats des del mateix procés.
- **Revisió de comandes internacionals:** L'enviament de comandes internacionals es fa des d'una empresa de transport diferent a les nacionals, i amb costos diferents. Justament per això és molt important que en el procés d'enregistrar les comandes s'hagi assignat correctament la nacionalitat del client. Actualment han d'alternar la consulta de l'aplicació "Servir Todo Tiendas" amb la de l'ERP per corregir possibles errades en la introducció de dades. Es vol integrar tot aquest procés també en l'aplicació "Servir Todo Tiendas", de forma que des de la mateixa aplicació pugin corregir les dades de contacte del client sense necessitar d'accedir a l'ERP.
- **Control i seguiment des de les botigues:** Es vol dotar d'un sistema unificat per la revisió i preparació del material de les botigues que han d'enviar cap a la central. En concret es vol unificar amb el mateix sistema que ja volen integrar per l'aplicació "Pedidos", de forma que des d'una sola pantalla puguin gestionar tant el material que han de reservar per les altres botigues (enviament setmanal), com pel magatzem central (enviament diari)
- **Extracció de material de traspassos d'estocatge:** Durant el cap de setmana es preparen un seguit de traspassos de material de la central cap a les botigues per emplenar l'estoc d'aquestes. Si es detecta un article necessari per a poder servir la comanda d'un client, el sistema hauria d'extreure aquest material del traspàs i assignar-lo a la comanda del client.
- **Execució controlada i independent:** Per tal d'evitar que el sistema quedi paralitzat per un possible error que es generi al tractar una comanda dins el conjunt de comandes a tractar, es vol que el sistema sigui capaç d'executar les comandes de forma independent. Si alguna d'elles produeix algun error, es recapitularà i es mostrarà en un resum final a l'usuari, però mai s'ha de paralitzar la resta de possibles comandes a servir.

Tenint en compte tots aquests punts anotats podem anotar a continuació un diagrama del flux de treball que haurà de dur a terme l'aplicació:





### Verificacions prèvies

- **Sèries invàlides:** Es llistaran els documents amb sèries invàlides i s'exclouran de l'objectiu de l'aplicació. L'usuari haurà de corregir-les manualment.
- **Estoc fictici:** Es llistaran els articles categoritzats com d'estoc fictici, i es permetrà a l'usuari assignar l'article substituït a través d'una cerca dins dels articles existents. Caldrà poder gestionar quins són els articles que anomenen "d'estoc fictici" i poder assignar l'article substituït per defecte.
- **Comandes internacionals:** Es llistaran totes les comandes considerades internacionals (es detectarà pel camp "PAIS" i "PROVINCIA" de la base de dades de l'ERP), perquè l'usuari pugui verificar si tots els que hi apareixen són realment internacionals. Si convé l'usuari podrà modificar les dades de la direcció del client per ajustar-les a la realitat o corregir-les.

### Configuració i parametrització

- **Magatzems disponibles:** Caldrà poder escollir quins magatzems entren dins el procés de càlcul, tant per la part del tractament de disponibilitat COMPLETA com la DISPERSA.
- **Prioritats dels magatzems:** Caldrà especificar una cadena de prioritats a l'hora d'escollir de quin magatzem extreure el material en cas de múltiple disponibilitat (interessa deixar abans amb menys estoc certs magatzems que d'altres).
- **Prioritat geogràfica:** Caldrà crear un sistema ajustable pel que fa a les prioritats d'enviament d'una comanda COMPLETA. Volem poder decidir per cada país i província, quina cua de prioritats té a l'hora d'escollir la disponibilitat d'estoc en múltiples magatzems.
- **Altres:** Cal poder ajustar altres elements en la configuració com poden ser: Recàlcul d'estocs prèvia execució del procés, considerar unitats en trànsit al consultar les disponibilitats, extreure material de traspassos d'estocatge,...

**Execució del procés:**

L'execució del procés després de les verificacions prèvies, requerirà una sola interacció de l'usuari, i es calcularà de forma automàtica la franja d'execució que li pertoca (matí o tarda), permetent però la selecció manual d'aquesta.

**Resultats del procés:**

- **Resum de resultats del procés:** Després de la finalització del procés, caldrà mostrar a l'usuari el resum de resultats (quantes comandes s'han servit, traspassos generats, ...) i qualsevol possible error que s'hagi generat durant l'execució (indicant en quina comanda, codi i missatge d'error).
- **Documents generats:** El procés genera els documents necessaris per repartir les següents tasques distribuïdes de l'empresa.

Els documents que cal generar, són:

- Resum d'albarans generats.
- Llistat d'albarans generats (un full per cada albarà en idèntic format que els utilitzats en l'ERP).
- Llistat de caixes a preparar al magatzem.
- Llistat de material extret de traspassos d'estocatge.
- Llistat detallat d'albarans amb disponibilitat parcial (només hi ha una part del material disponible).
- Llistat de canvis d'articles d'estoc fictici que s'han corregit, amb la informació de la comanda, telèfon del client, ...
- Llistat de traspassos generats de les botigues cap a la central, amb els corresponents articles involucrats.
- Llistat d'albarans amb comentaris o personalitzacions (s'utilitzen uns codis d'article especials per aquesta funció) per a revisar abans de l'enviament, ja que pot suposar contactar amb el client per qualsevol aclariment.
- Llistat d'articles a preparar al magatzem per a totes les comandes, ordenats per la ubicació física de l'article, per permetre la recollida del material d'una forma òptima i àgil.

**Faltes d'estoc**

Igual que en la primera aplicació "Pedidos" vista anteriorment, actualment s'utilitza l'ús d'una referència anomenada "DOBAR" per a indicar la falta d'estoc d'algun article. Com en el cas anterior, haurem d'utilitzar un sistema de faltes que en permeti l'adequada gestió i control d'aquestes.

**Balanceig d'estocs**

Al escollir d'on hem d'extreure el material per a servir les comandes al client, haurem d'aplicar un algorisme que respecti el balanceig d'estocs, de forma que apliqui un consum d'estocs distribuït, evitant deixar un magatzem sense quan n'existeix disponibilitat sobrant en d'altres.

### 5.6.3 Aplicació “Pendientes” i “Pendientes Y3” (Bloc 1 – Logística)

#### Canvis en l'estructura de les aplicacions

El client planteja dos objectius primordials en l'ús d'aquestes dos aplicacions:

- **Unificar-les:** Crear una sola aplicació que permeti consultar indistintament les comandes que s'han d'enviar a client final, així com les que es recullen directament a la botiga. Per tant de la suma d'aquestes dos aplicacions n'apareixerà una de sola anomenada **“Pendientes de Servir”**
- **Dades actualitzades en tot moment (on-line):** Fins ara ambdós aplicacions contenien dades actualitzades de forma diària (Aplicació “Actualizar Pendientes Servir”), o sigui eren *“off-line”*. Es desitja que totes les dades siguin completament síncrones amb les bases de dades de l'ERP i per tant estiguem visualitzant les dades actuals al moment (*on-line*).

#### Ampliació de funcionalitats

A part de les funcionalitats existents en l'aplicació actualment, es vol dotar a la nova aplicació de funcionalitats addicionals:

#### Indicadors d'estat de la comanda

Caldrà fer ús d'uns indicadors visuals de ràpida interpretació que ens permeti tenir informació global de la comanda i de les seves línies de forma independent també. En concret volem disposar d'indicadors per diferents categories:

- Estat de la comanda: Disponibilitat parcial, total o sense disponibilitat.
- Estocs en trànsit: Existeixen unitats de camí (en transport)
- Comandes fetes al proveïdor: Existeixen unitats demanades al proveïdor o al magatzem central.
- Consultes realitzades sobre la comanda: Consultes associades a la interacció amb el client o internament per comunicar dates d'entrega, trucades pendents, notes pendents de llegir,...

#### Llistat de comandes navegable, complet i accessible

Es vol disposar d'un llistat que sigui fàcil, còmode i complet pel que fa a les diferents comandes i de les línies d'articles de les mateixes, de forma que al seleccionar una de les comandes s'accedeixi fàcilment a les seves línies, i una vegada accedim a la línia s'accedeixi també fàcilment a les dades de l'article que la representa.

En concret es demanen les següents funcionalitats en dit llistat:

- Es pugui ordenar pels camps data, número de sèrie, número de document i nom del client.
- Poder imprimir un llistat (detallat o no) del llistat de comandes que s'estigui visualitzant.
- Poder filtrar el llistat actual de comandes per diferents criteris:

1. Per l'estat de la comanda.
  2. Per l'existència o no de consultes associades.
  3. Per un article en concret.
  4. Per la marca d'un article en concret
  5. Segons si s'ha processat o no la comanda a través d'algunes de les aplicacions bàsiques de gestió de comandes ("Pedidos" o "Servir Todo Tiendas").
  6. Per un interval de dates.
- Poder cercar totes les comandes històriques d'un client.

### **Informació agregada a la comanda**

També es vol disposar de fàcil accés a informació extra de la comanda o les seves línies, com pot ser:

- Dades del client
- Existència d'avançaments en el pagament de la comanda.
- Dades de l'article (Sobretot referent a l'estoc)
- Compres pendents de rebre del proveïdor.
- Resum d'estocs de l'article pendent per magatzems.
- Relació d'altres comandes que tenen pendent el mateix article de la que estem visualitzant.

### **Consultes associades a les línies de comandes pendents**

Caldrà poder associar consultes a cada una de les línies de material pendent d'una comanda, o sigui a cada article/s pendent de servir al client.

Quan parlem d'una consulta la considerem com un conjunt de converses on diferents usuaris poden anar anotant observacions i marcant diferents fites informatives en el procés de comunicació amb el client final o en l'estat de la comanda, com poden ser:

- Petició d'informació genèrica.
- Necessitat de trucar a un client.
- Marcar que s'ha trucat al client i s'hi ha pogut parlar.
- Intent de contactar amb el client infructuós.
- Deixat missatge al contestador del telèfon del client.
- Esperant la resposta d'un client.
- Cal anular la comanda.

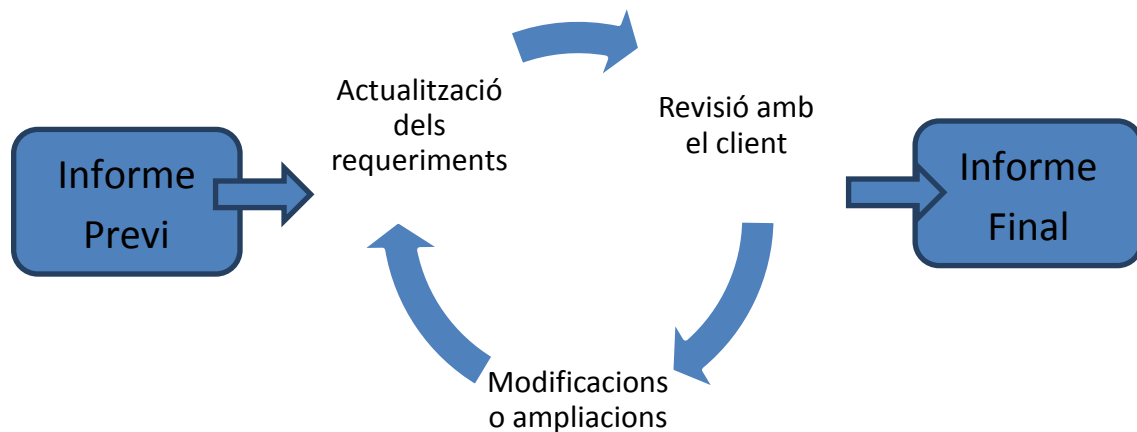
Al crear una entrada o conversa dins la consulta, caldrà enregistrar:

- Data i hora de la nova entrada
- Origen i destí de la conversació (Botiga1 -> Botiga2, Botiga5 -> Magatzem, Magatzem -> Botiga1, ...)
- Usuari que genera la conversa
- Tipus (alguna de les possibles fites informatives anotades anteriorment)
- Si s'ha llegit o no per part del destinatari.

## 5.7 Informe previ i revisió de l'anàlisi de requeriments

Una vegada recaptada tota la informació referent a l'anàlisi de requeriments de les aplicacions existents així com noves funcionalitats que es vulguin implementar, es presenta un informe previ al client amb els continguts de l'apartat 5.6.

S'estableixen les corresponents reunions per a l'estudi de l'informe amb el client, deixant obert el cicle de vida de l'anàlisi fins que es determini en la seva totalitat les necessitats actuals i reals del sistema desitjat.



Una vegada obtingut l'informe final, entrem dins la fase del disseny tècnic i funcional.

## 6. Especificació i disseny

En aquest apartat ens centrarem a especificar i dissenyar l'arquitectura del sistema SetelSys, per complir amb els requisits detectats i desitjats.

Tot seguit analitzem els models de dades existents ja en el sistema actual prèvia introducció del sistema SetelSys i amb els quals el nou sistema s'haurà de comunicar tant per la extracció com introducció de dades.

### 6.1 Models de dades existents en el sistema

Per tal de poder tenir una imatge semànticament fiable de les bases de dades existents en el sistema, tant pel que fa a l'ERP com a la pàgina WEB comercial, hem utilitzat un procés d'**enginyeria inversa** per poder plasmar el model entitat relació existent en les bases de dades actuals de dits components.

Analitzem en tot moment únicament les entitats relacionades amb l'objectiu de la nostra aplicació deixant a part tota la resta d'entitats de la base de dades que no ens incumbeix.

#### 6.1.1 Model de dades de l'ERP principal

##### Comandes de venda

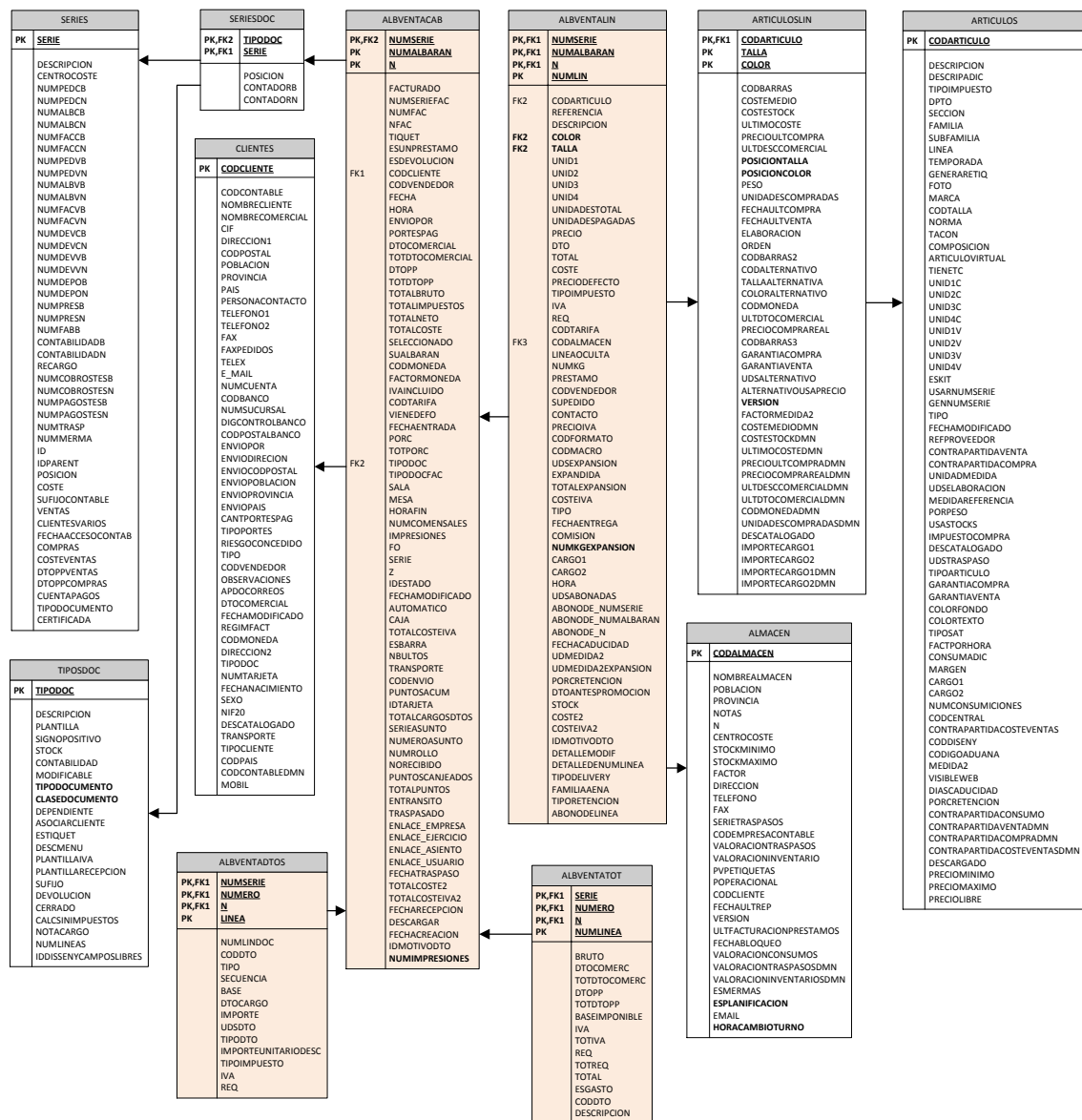
Les taules centrals d'aquest model són, **PEDVENTACAB** i **PEDVENTALIN**, que s'utilitzen per emmagatzemar la informació de la capçalera i línies de la comanda. Existeixen un seguit de taules fruit de la relació entre alguns dels seus camps i altres taules de l'ERP:

- **SERIESDOC**: Associa el document a la sèrie i tipus concrets per cada element. A la vegada aquesta taula es relaciona amb **SERIES** i **TIPOSDOC**.
- **CLIENTS**: Associa el document a un client concret.
- **ARTICULOSLIN**: Associa la línia del document a un article concret del sistema (Utilitzant la clau primària corresponent – Codi + Talla + Color). A la vegada aquesta taula depèn d'**ARTICULOS** on s'emmagatzema la informació genèrica dels articles (Referència, marca, departament, etc.)
- **ALMACEN**: Associa la línia del document a un magatzem existent. Això és necessari per que el sistema sigui conscient d'on s'extreu el material.



## Albarans de venta

Les relacions en aquest bloc de taules són les mateixes que en les comandes de venta, però utilitzant les entitats relacionades amb els albarans en comptes de les comandes, us les indiquem amb una tonalitat **especial** per visualitzar les úniques diferències.

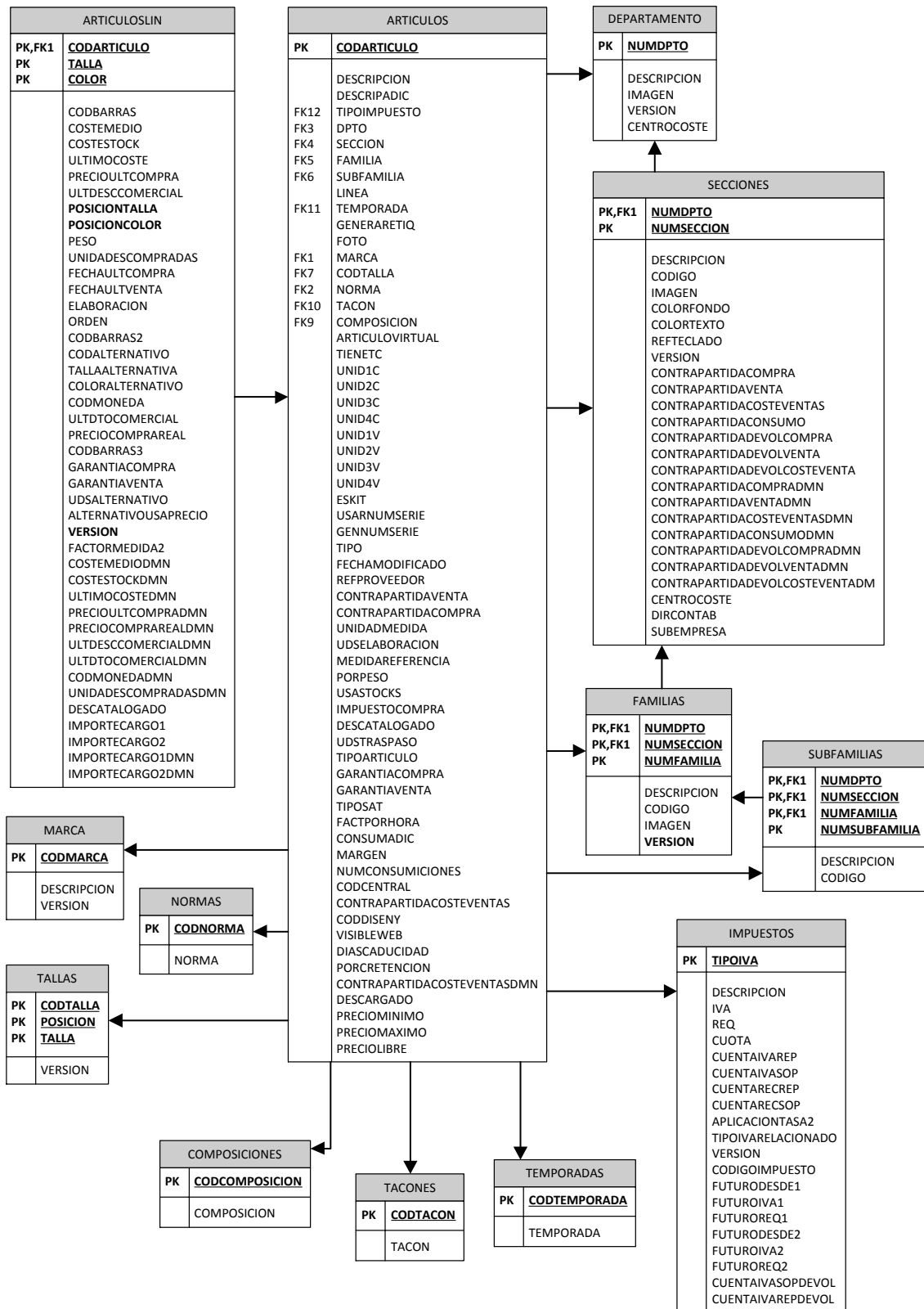




## Articles

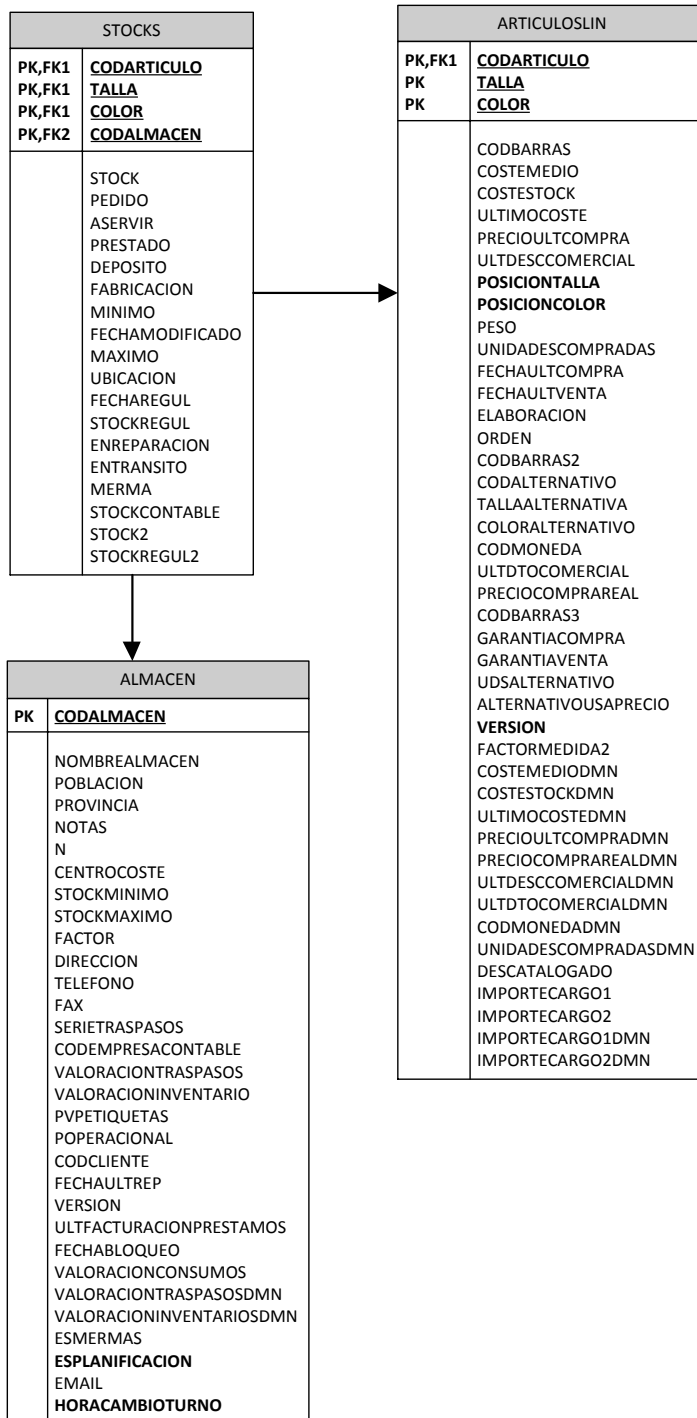
La taula principal d'articles és la **ARTICLES**, de la qual depèn una relació de composició amb **ARTICULOSLIN** (on per cada article s'especifiquen les diferents combinacions de talles i colors). A la vegada la taula **ARTICLES** conté atributs que es relacionen com a claus foranies amb altres taules del sistema:

- **DEPARTAMENTO, SECCIONES, FAMILIAS, SUBFAMILIES**: Taules utilitzades per a la classificació categòrica de l'article i que s'utilitzen per definir la classificació de tots els articles de la base de dades. La relació però no té perquè ser completa en els 4 nivells, o sigui que poden existir articles associats a un departament, però no a cap secció, família o subfamília, o poden existir articles amb tots els nivells de categorització assignats.
- **IMPUESTOS**: Per definir el tipus d'impost associat a un article.
- **TALLAS**: Es referencia aquesta taula per assignar el codi de tallatge associat a l'article.
- **MARCA**: Per assignar la marca de l'article a una de les existents al sistema.
- **NORMAS, TACONES, TEMPORADAS i COMPOSICIONES**: Utilitzades per categoritzar els detalls de constitució de l'article, la seva funcionalitat o composició.



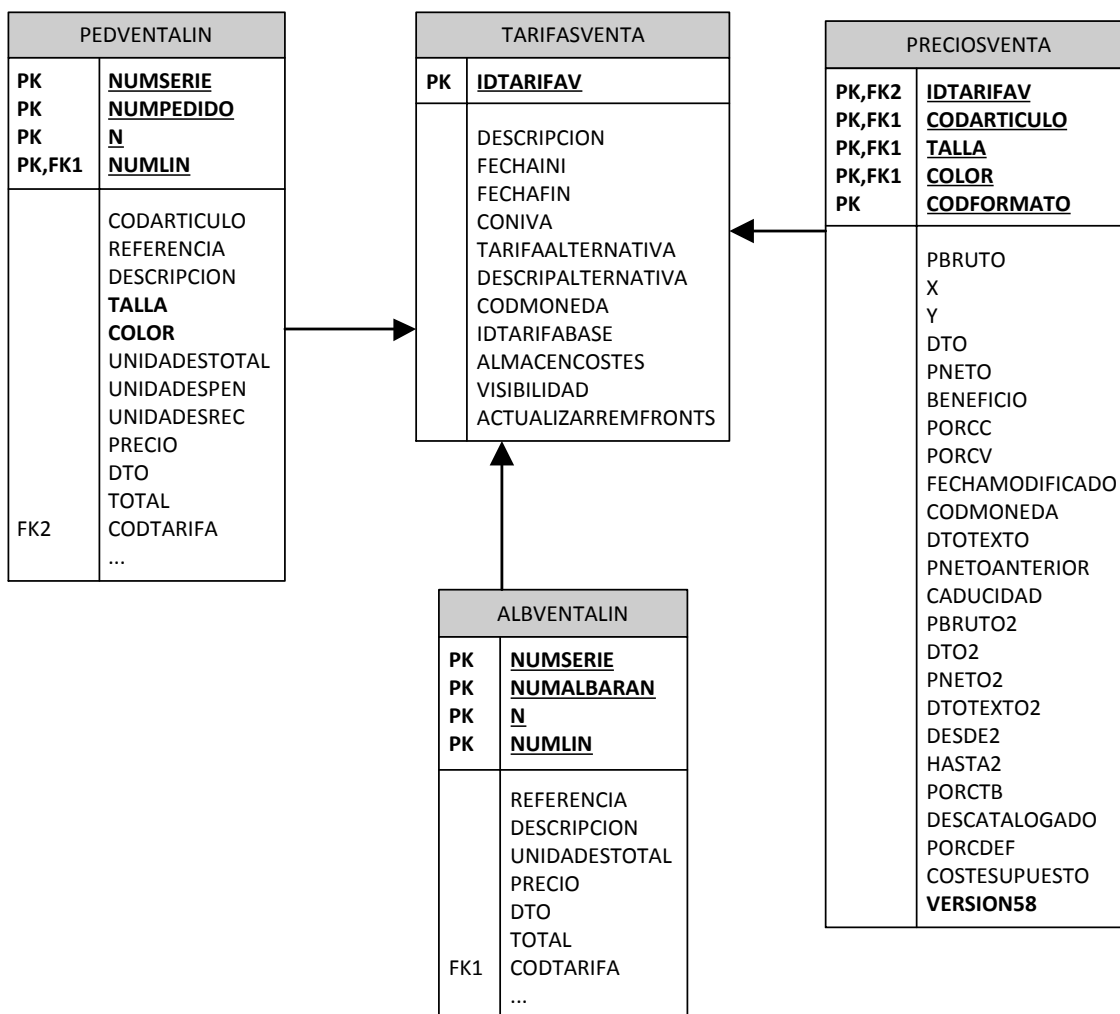
## Estocs

Aquest bloc del model es basa en l'existència d'una taula **STOCKS**, que es relaciona amb **ARTICULOSLIN** i **ALMACEN**. Això permet definir, per a cada combinació d'article (codi, talla i color) i cada magatzem, les seves disponibilitats i estats d'estoc.



## Preus i tarifes de venta

La taula principal de referència és **TARIFASVENTA**, a la qual els diferents documents de venta, ja siguin comandes o albarans (a través de **PEDVENTALIN** o **ALBVENTALIN**) es relacionen per assignar una tarifa associada a la venta (Utilitzant el camp **CODTARIFA**). A la vegada tots els preus existents dels articles es relacionen amb la taula **TARIFASVENTA** com anota el bloc del model que mostrem a continuació. Això permet que podem establir tots atributs de **PRECIOSVENTA** per cada combinació existent d'article (codi, talla i color) i tarifa (a través del seu identificador enter).



### 6.1.2 Model de dades de la web comercial

La base de dades de la web corporativa s'utilitza bàsicament per dos objectius:

- Permetre una representació intermèdia de les entitats que es requereixen per la comercialització: Articles, preus, estocs,... I no dependre d'una connexió real amb el sistema. Això implica un procés d'actualització diari d'aquestes dades, que no ens afecta al nostre sistema ja que la gestió de dit procés està a càrrec de l'empresa desenvolupadora de la web.
- Permetre emmagatzemar la informació necessària pel correcte funcionament de l'aplicació web comercial (usuaris, imatges, seccions, cistella de la compra, ...)

No obstant, existeixen alguns processos que en un futur no molt llunyà l'empresa voldrà que es gestionin des del mateix SetelSys. Justament per això, i tenint en compte que ja hi ha un procés que requereix de la connexió amb la base de dades de la web, aprofitem aquest apartat per veure parts del model de dades de la web que ens farà falta conèixer i controlar per futures ampliacions de funcionalitats.

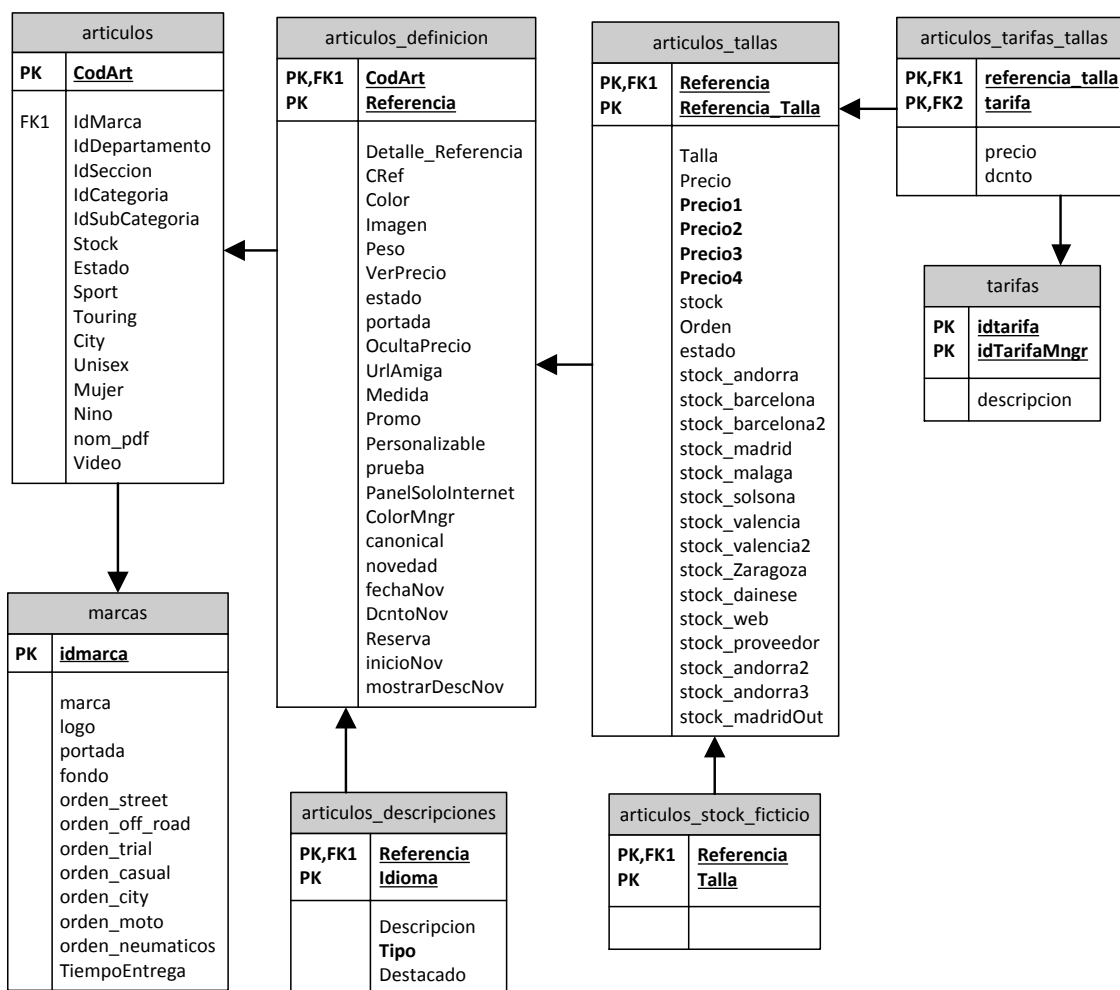
Així com l'ERP principal disposa d'una base de dades semànticament més ben construïda i en una forma correcta, no podem dir exactament el mateix de la pàgina web, en la que detectem certes irregularitats i un estat no precisament normalitzat.

## Articles i preus associats

La taula principal d'aquest bloc és la de *articulos*, on s'emmagatzema la informació bàsica i categòrica de l'article (marca, departament, secció, família, etc.). És important denotar que en aquesta base de dades, un article

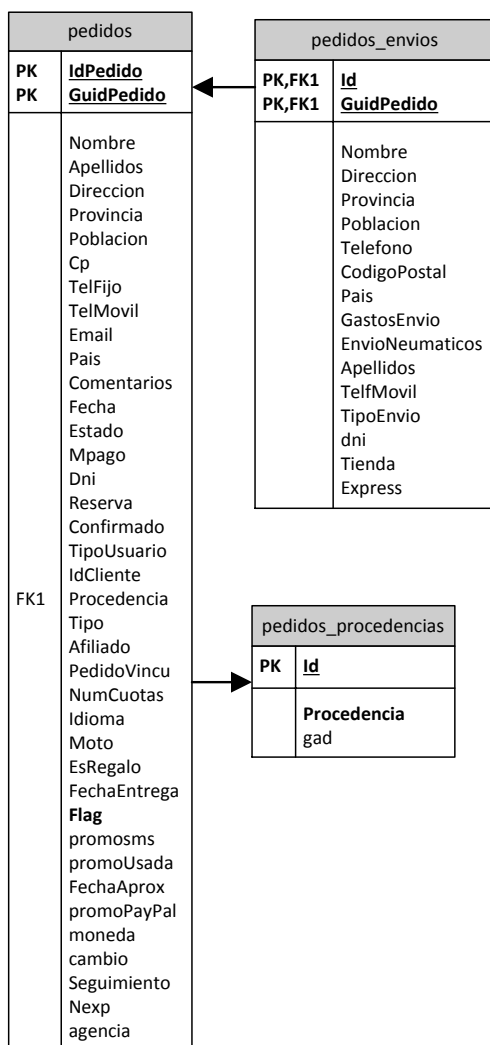
La taula *articulos\_definicion* es relaciona amb *articulos*, bàsicament per ampliar la informació associada a l'article de forma general (referència dins l'ERP, pes, estat, si és o no novetat,...) i la taula *marcas* relacionada a través de *idMarca*. D'aquesta taula *articulos\_definicion* en depenen.

- *articulos\_tallas*: On es defineix la descomposició d'un article en les seves diferents talles. Amb la qual es relaciona *articulos\_tarifas\_tallas* i *tarifas* (que contenen la informació dels diferents preus assignat a l'article en les diferents tarifes).
- *articulos\_descripciones*: On s'assigna la descripció de cada article, en cada un dels possibles idiomes existents en la web.



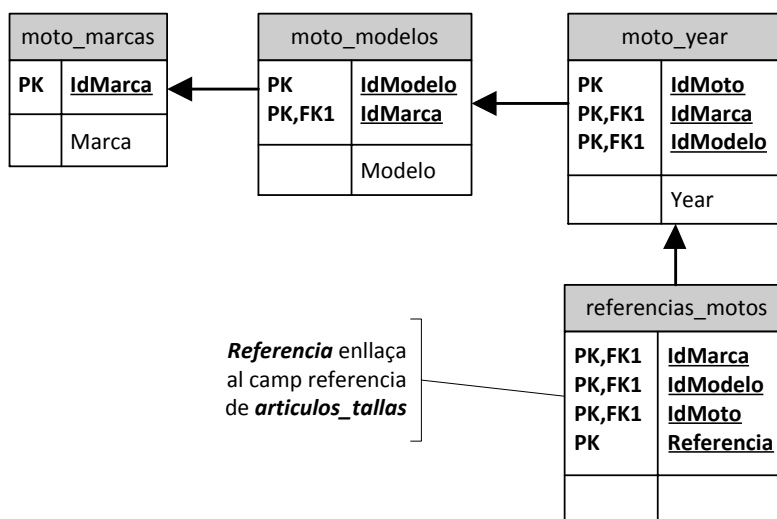
## Comandes generades a la pàgina web

Les comandes que es generen dins la pàgina web s'emmagatzemen a la taula **pedidos**, a la qual s'hi relaciona **pedidos\_envios**, amb la informació de l'enviament de la comanda (dades d'enviament, tipus d'enviament, magatzem origen de l'enviament,...). També la taula **pedidos** es relaciona amb **pedidos\_procedencias** per controlar l'origen de la pàgina web o webs que l'enllacen, des d'on s'ha iniciat la comanda.



## Gestió de motocicletes i articles associats

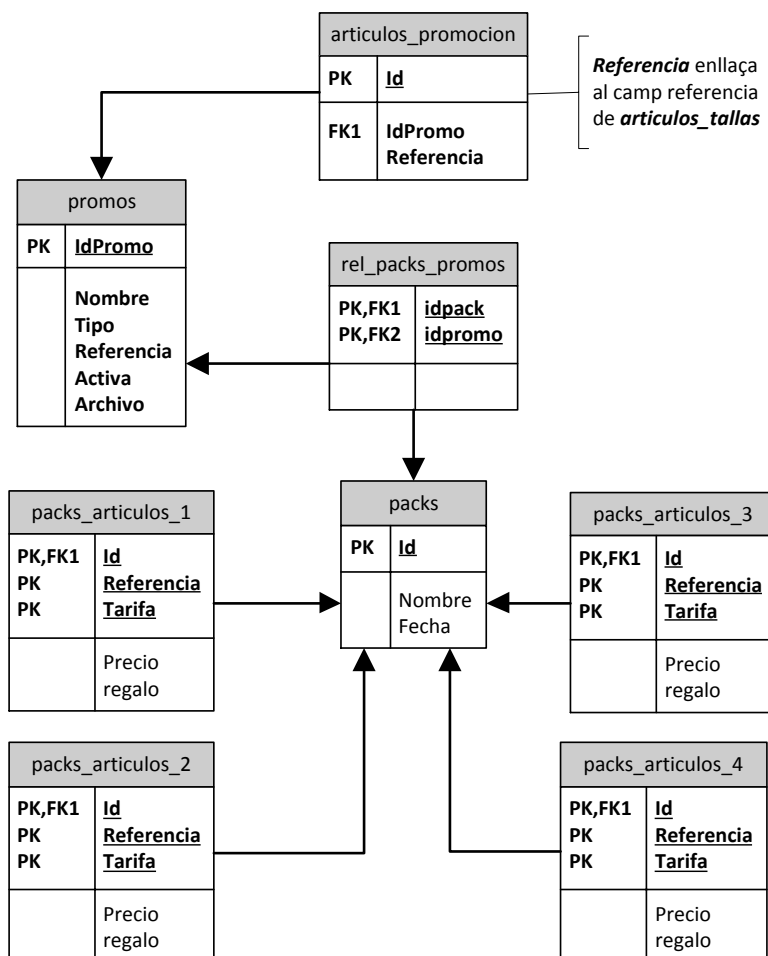
Des de la web es gestionen els articles associats als diferents models de motocicletes, aquestes es distribueixen en una cadena de relacions d'agregació, de forma que tenim la taula principal ***moto\_marcas*** que es compon de ***moto\_modelos*** i aquesta a la vegada de ***moto\_year***. Existeix una taula, ***referencias\_motos***, on s'associen els articles a les motocicletes existents a ***moto\_year*** i a la vegada es relaciona amb la taula ***articulos\_tallas***:





## Gestió de packs

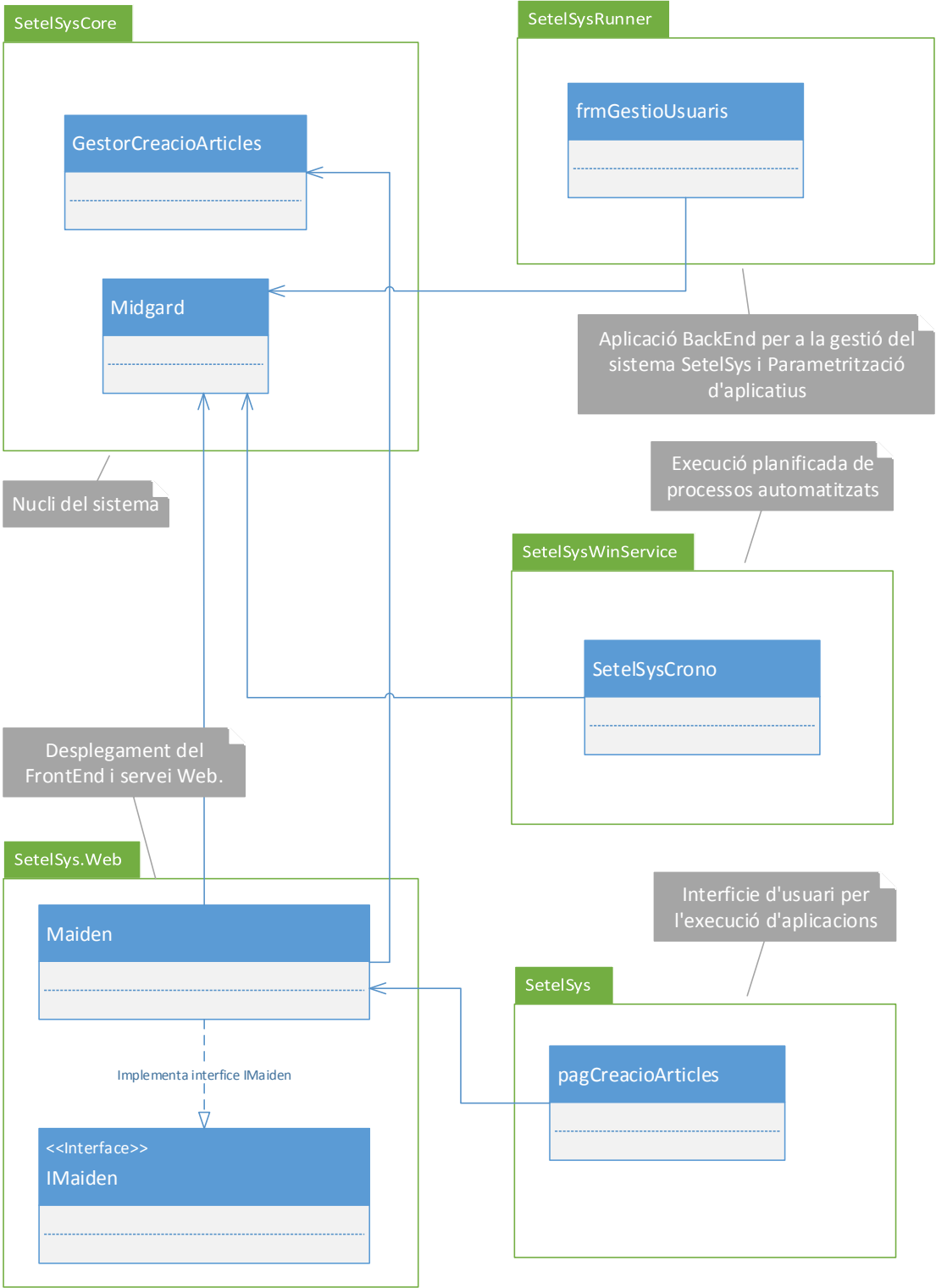
En la promoció de la venda d'articles a la pàgina web, aquests es registren a la taula de **packs** per permetre publicar ofertes especials. En concret de la taula **packs** es relacionen quatre taules amb combinacions diferents d'articles per aquell pack (**packs\_articulos\_x**). A la vegada existeix una taula de control **rel\_packs\_promos** que relaciona els diferents **packs** existents amb promocions representades a través de la taula **promos** on s'hi relacionen els articles que la conformen a través de **articulos\_promocion**.



# 6.2 Especificació de l'arquitectura del sistema SetelSys

## 6.2.1 Diagrama de components i dependències del sistema

Mostrem a continuació un diagrama del disseny dels diferents components del sistema i les dependències existents entre ells. Posteriorment fem una descripció dels diferents components.



## SetelSysCore

Nucli del sistema encarregat de gestionar tota l'operativa funcional amb:

- Connexió física amb les bases de dades. (A través de la classe **SQLServer**)
- Representació i gestió de l'operativa bàsica CRUD associada a les classes representatives de les entitats existents en el sistema ERP existent i la web comercial vinculada. (**WebDataCom** i **ERPDataCom**)
- Representació i gestió de l'operativa bàsica CRUD associada a les classes del sistema SetelSys (**SetelSysDataCom**).
- Gestió d'errors del sistema i missatges textuais associats a dits errors, així com la gestió de generació d'enviaments de correus i missatgeria (**Mercury**).
- Implementació dels diferents gestors d'aplicacions, encarregats d'executar les tasques especialitzades demanades per cada aplicació.
- Implementació d'un controlador especialitzat en servir les connexions i accés als diferents components del sistema (**Midgard**).

## SetelSysRunner

Interfície realitzada amb WinForms, encarregada d'implementar l'operativa corresponent a:

- Gestionar els usuaris, botigues, permisos, aplicacions, ...
- Gestionar les parametritzacions especialitzades requerides per algunes de les aplicacions o funcionalitats desitjades pel client.
- Gestionar les aplicacions automatitzades actualment vigents en el sistema així com la seva parametrització i patrons d'execució.
- Visualitzar el registre d'activitat de les aplicacions automatitzades.
- Eines per l'agilització del manteniment.
- Incorpora totes les classes referents a formularis d'interacció gràfica amb l'usuari administrador del sistema **SetelSys**.

## SetelSysWinService

Servei de Windows encarregat de l'execució en segon pla de totes les aplicacions automatitzades segons el patró que tenen establert. Utilitza la classe **SetelSysCrono** per a formalitzar la interacció amb el nucli (**SetelSysCore**) i obtenir la informació de les aplicacions vigents, paràmetres i temporització. L'interval de verificació d'execucions és cada 1 minut i s'inicia de forma automàtica a l'iniciar el sistema (independentment de l'inici de sessió de cap usuari), utilitzant la compta d'usuari local del sistema.

**SetelSys.Web**

Component encarregat de carregar la interfície gràfica d'usuari elaborada amb Silverlight i que utilitzaran els usuaris consumidors del sistema, a diferència dels administradors que tindran les tasques centralitzades a través del **SetelSysRunner**.

Dit component allotja també la definició del servei web **Maiden**, encarregat de connectar-se amb el **SetelSysCore** per efectuar les operacions necessàries per les diferents aplicacions utilitzant classes heretades de **GestorAplicacio**.

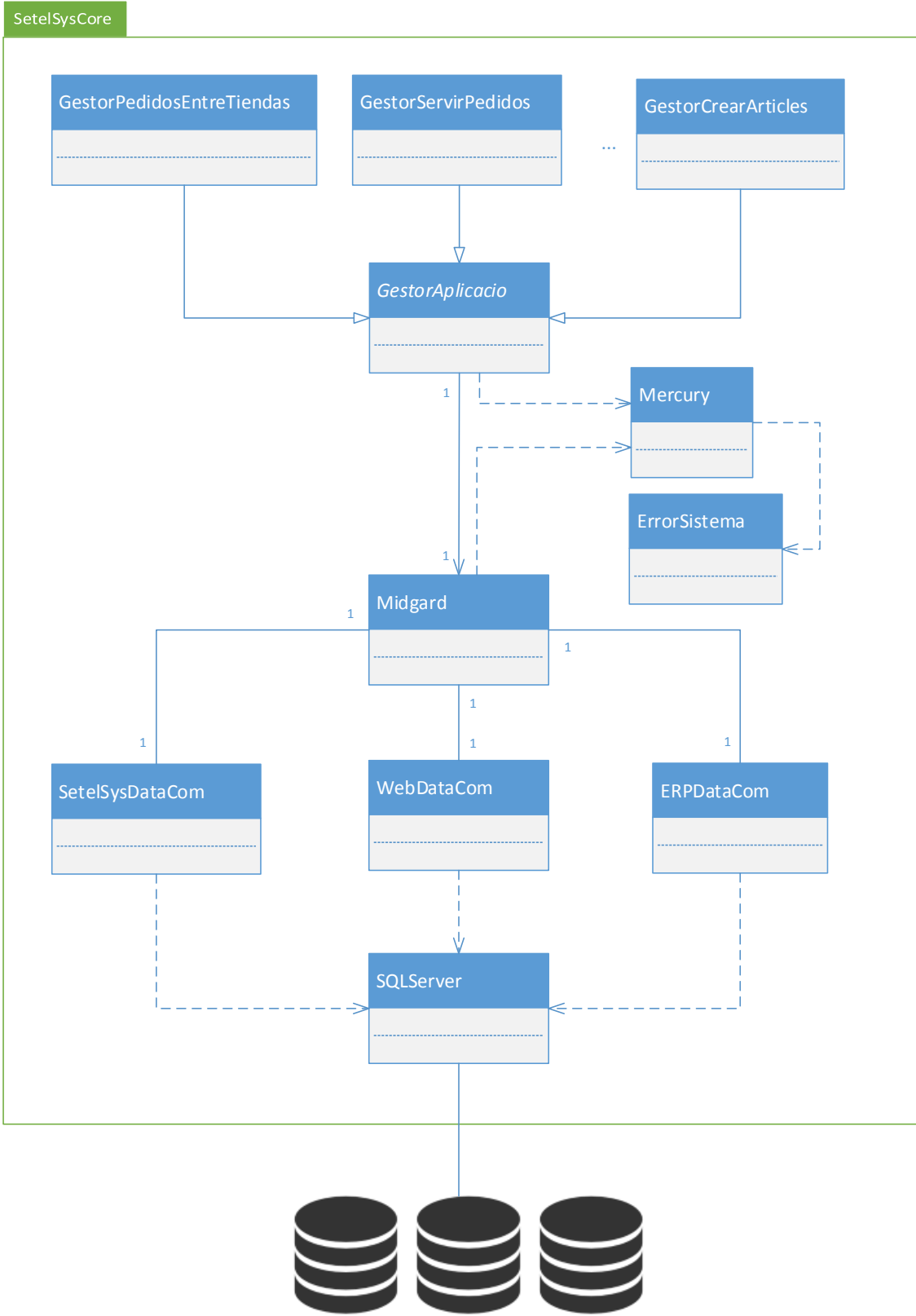
**SetelSys**

Interfície d'usuari elaborada amb Silverlight, encarregada de mostrar les pantalles d'interacció amb l'usuari consumidor. El component s'estructura en pàgines i finestres de Silverlight a través d'un menú principal generat dinàmicament en base dels permisos de l'usuari que es connecta pel que fa a l'execució de les aplicacions disponibles.

Tota interacció de dades entre la interfície i el nucli principal s'efectuen a través de la crida del servei **Maiden** allotjat en el component anteriorment descrit **SetelSys.Web**.

6.2.2 Diagrama de classes del nucli del sistema (SetelSysCore)

Com es podrà observar en el següent diagrama, el nucli s'estructura a través de les classes següents:



Les diferents classes que hereten de **GestorAplicació**, s'utilitzen per a mapejar les funcionalitats assignades a les diferents aplicacions del sistema. Disposen d'accés a la classe **Midgard**, encarregada de gestionar els recursos del sistema i la qual té una instància de les diferents classes de funcionalitat bàsica encarregats de la comunicació amb les dades finals a través de la classe **SQLServer**. Existeix un comunicador de dades per cada element existent al sistema i pel propi sistema SetelSys dissenyat:

- **ERPDataCom**: Encarregat de gestionar la funcionalitat d'accés i representació de les diferents entitats de l'ERP.
- **WebDataCom**: Encarregat de gestionar la funcionalitat d'accés i representació de les diferents entitats de la web comercial.
- **SetelSysDataCom**: Encarregat de gestionar la funcionalitat del sistema de control annex necessari per un correcte funcionament i compliment de requisits encarregats en l'anàlisi de requeriments.

Finalment cal destacar la dependència existent entre les classes GestorAplicacio i Midgard, amb la classe Mercury. L'objectiu de la qual ja hem esmentat anteriorment i es pot resumir en la gestió dels errors detectats en el sistema i de la gestió multilingüe futurament necessària, així com la gestió d'enviaments de correu i missatgeria en general.

### 6.3 Especificació del gestor del sistema (SetelSysDataCom)

Aquest component, com hem dit abans s'encarregarà de la gestió de les necessitats i infraestructura pel correcte funcionament del sistema de suport per l'execució d'aplicacions, gestor de permisos i en general la gestió de les diferents entitats i classes necessàries.

A continuació mostrem un anàlisi dels casos d'ús detectats per l'elaboració del disseny del sistema:

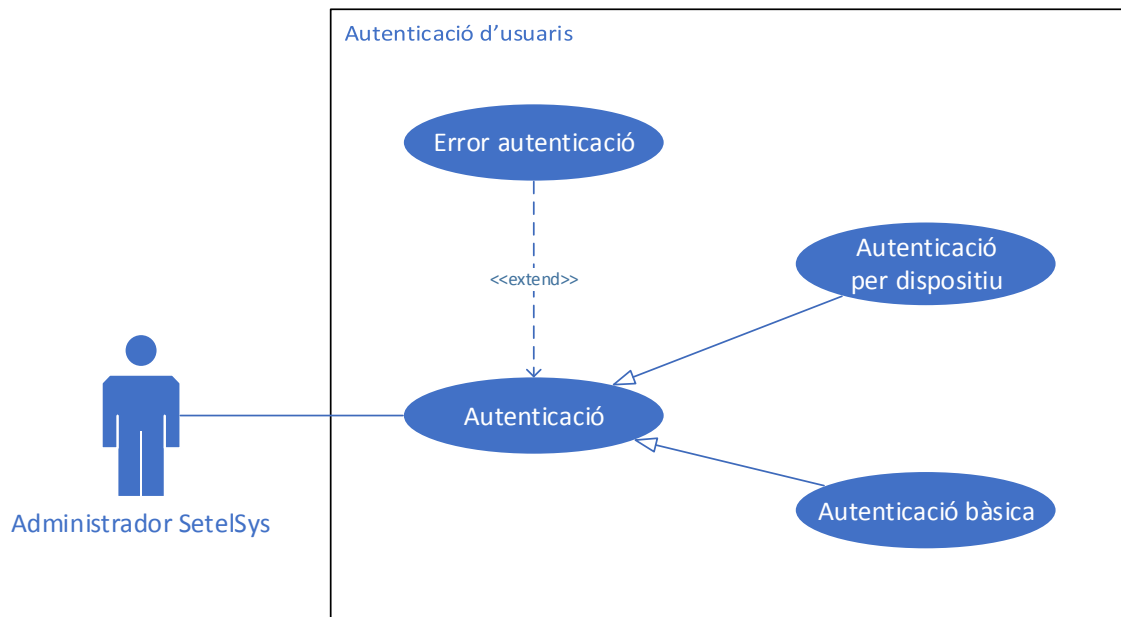
#### 6.3.1 Casos d'ús

##### Autenticació d'usuaris

*Qualsevol usuari, tant a nivell administratiu com estàndard caldrà que es validi en el sistema per acreditar-se com a autoritzat i poder disposar dels aplicatius disponibles pel seu usuari. En concret es poden donar 2 possibles vies d'autenticació:*

- *Autenticació bàsica: Basada únicament en un nom d'usuari i contrasenya.*
- *Autenticació per dispositiu: A part de l'usuari i contrasenya, caldrà validar si l'origen de la connexió està autoritzat.*

*El tipus de validació de l'usuari aplicada serà en base de la configuració general del sistema. En cas de produir-se un error d'autenticació caldrà notificar-ho degudament a l'usuari involucrat.*

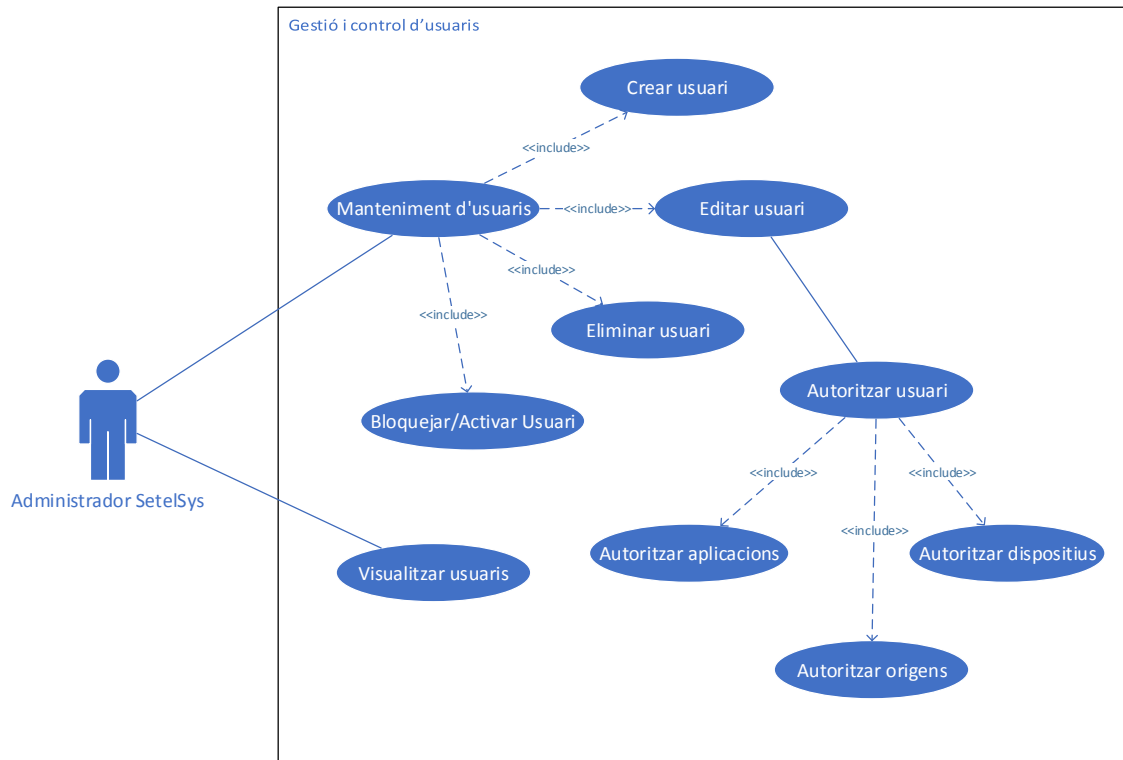


En els següents casos d'ús, per facilitar la lectura, suposarem que tota interacció es realitza sota el corresponent mecanisme d'autenticació.

### Gestió d'usuaris

Cal dotar al sistema de la capacitat de gestionar els usuaris que es podran acreditar per accedir al sistema. Caldrà poder llistar i aplicar el manteniment sobre qualsevol usuari, i també poder-li assignar la corresponent autorització de connexió en base de:

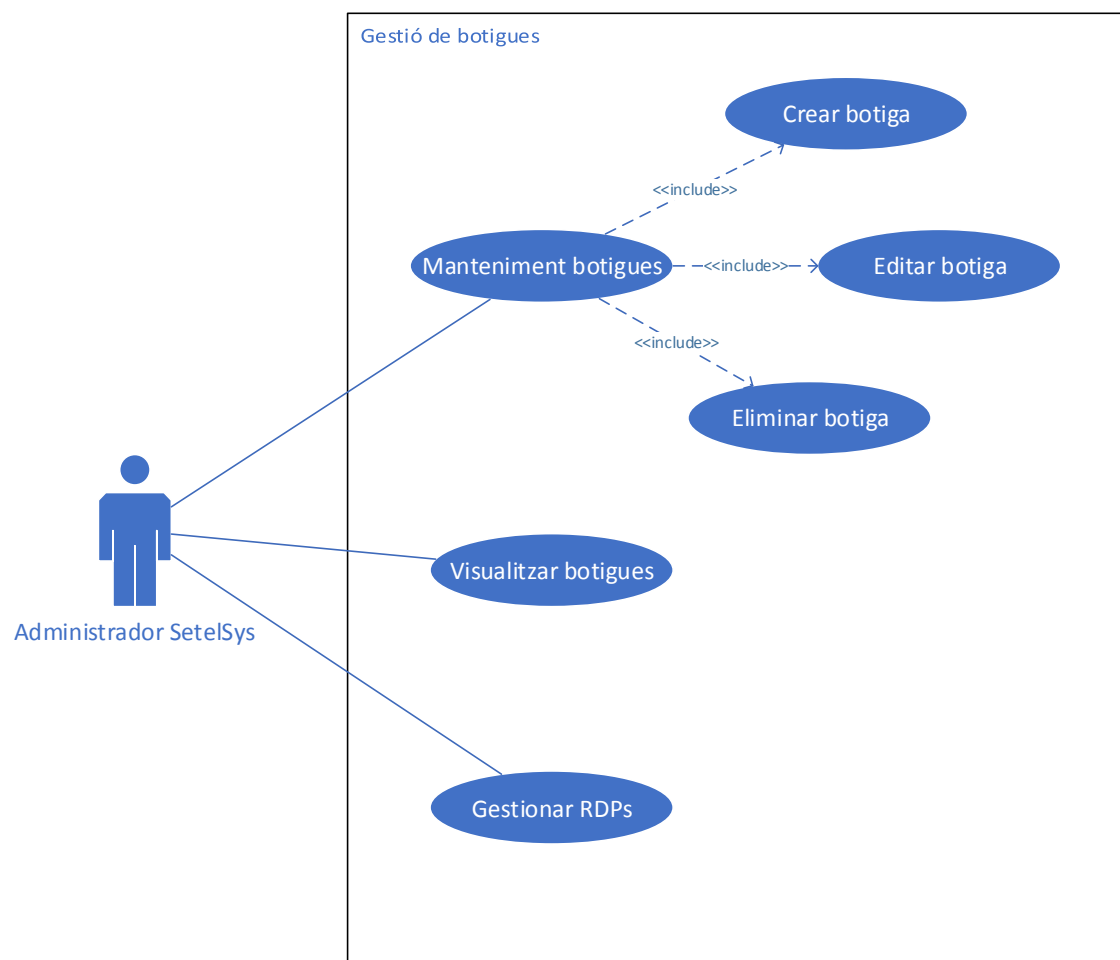
- Aplicacions autoritzades
- Dispositius d'origen de la connexió
- Botiga o central logística des de d'on s'inicia la connexió.





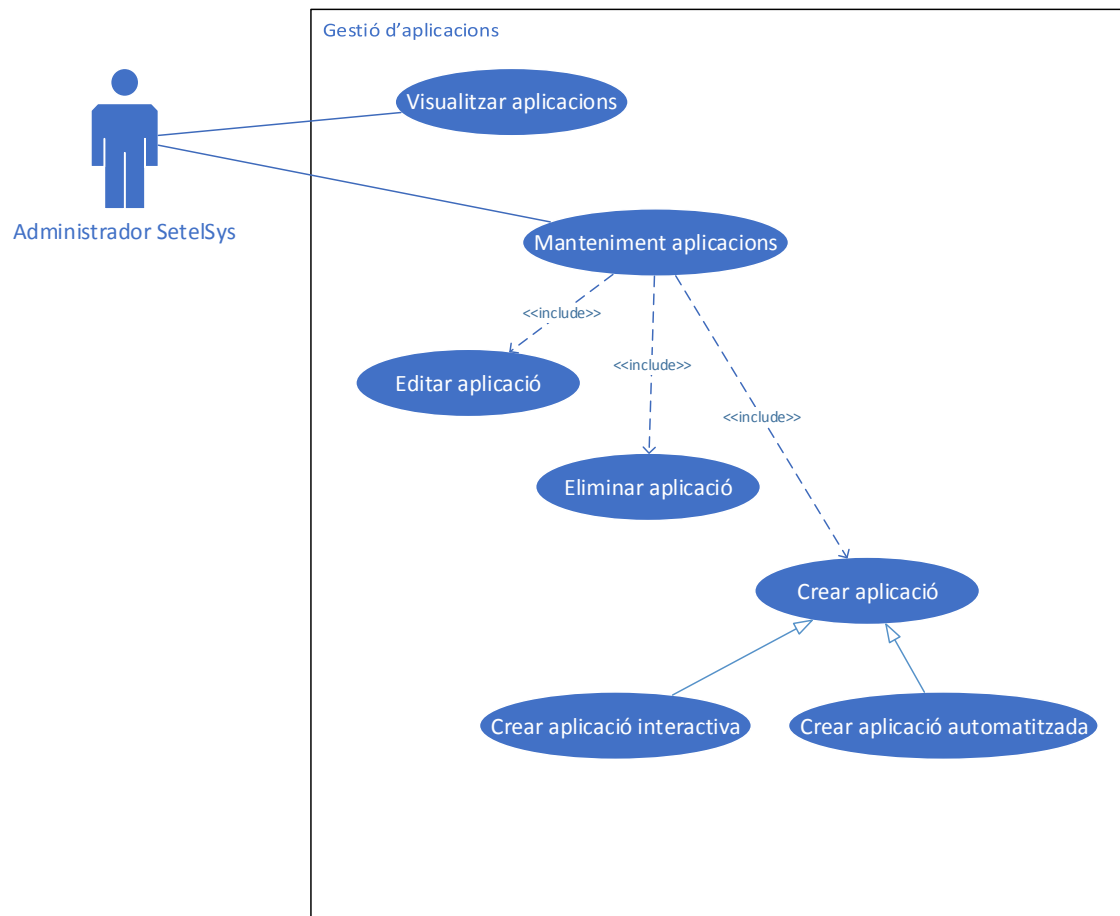
### Gestió de botigues

Caldrà poder llistar, gestionar i mantenir les diferents botigues registrades en el sistema i que s'utilitzaran com origen de connexió i validació de l'acreditació. També caldrà dotar al sistema dels recursos necessaris per poder assignar RDPs (Remote Desktop Protocol) a les botigues en qüestió per que el usuaris puguin descarregar-se dit fitxer generat de forma dinàmica des de l'aplicació.



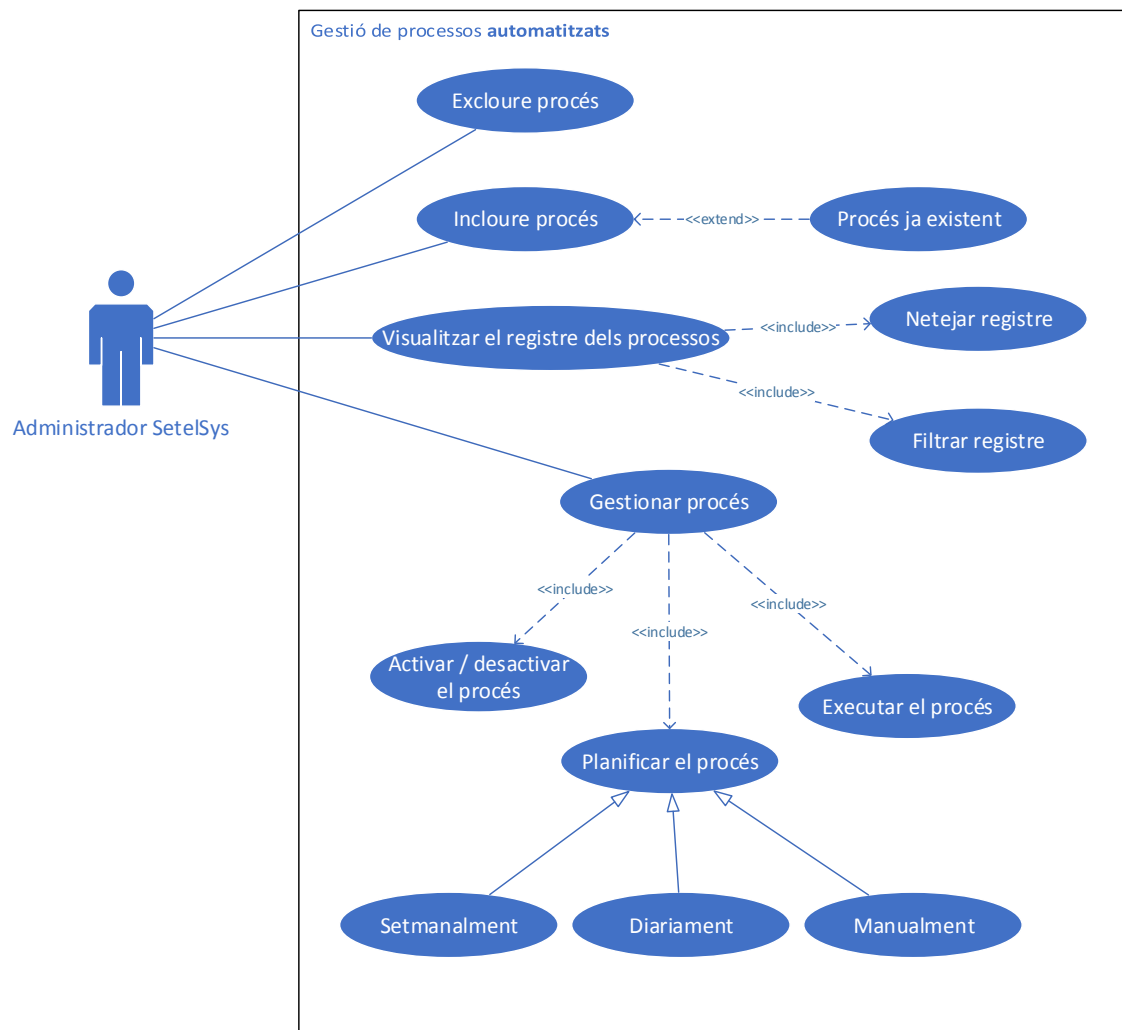
## Gestió d'aplicacions

Caldrà poder llistar les diferents aplicacions existents al sistema, així com gestionar-les i poder-ne crear de noves, ja siguin aplicacions interactives o bé aplicacions automatitzades (sense interacció amb l'usuari ja que tot el procediment està parametritzat per la definició dels seus objectius i es pot executar sense parametrització de l'usuari)

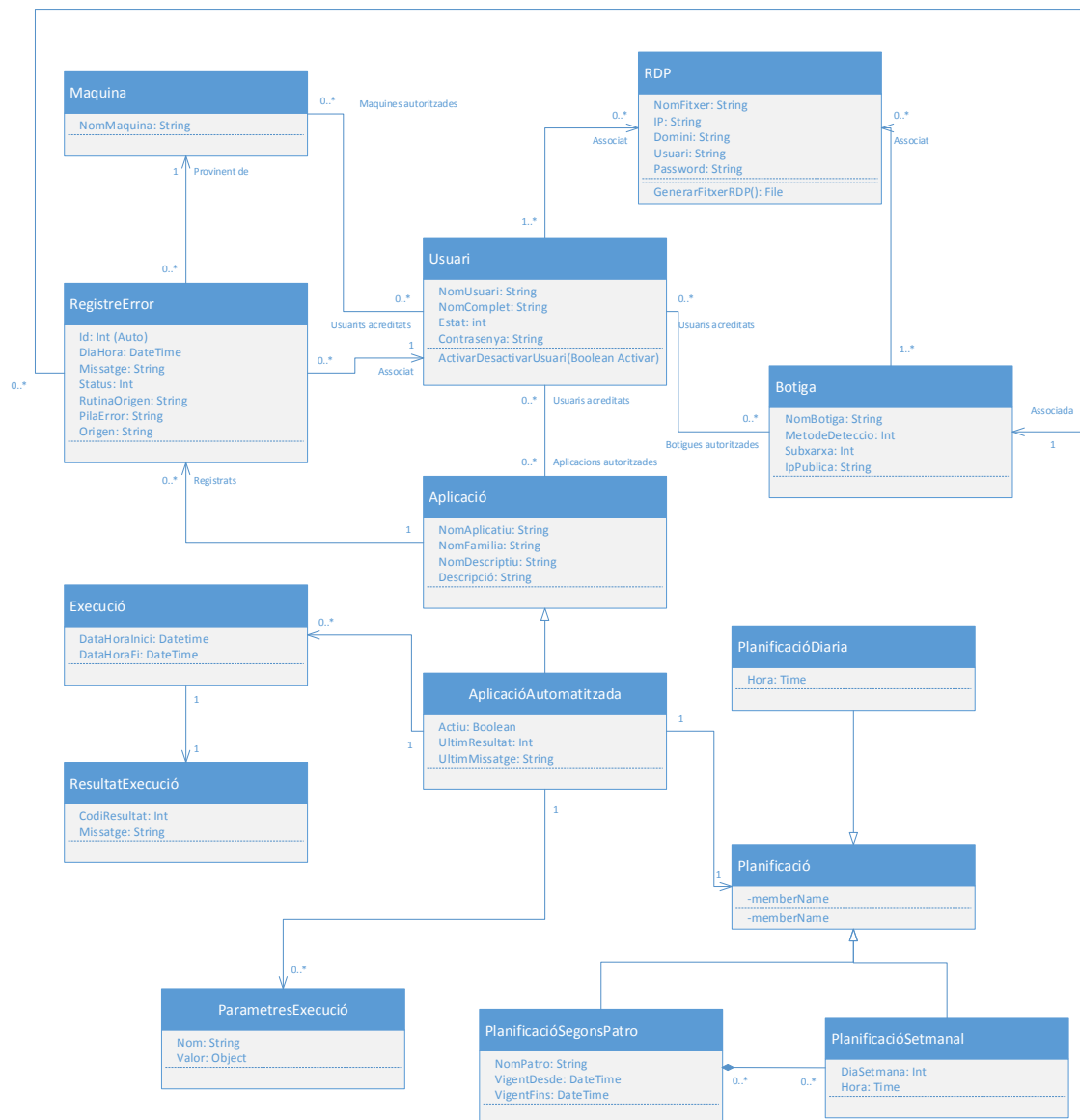


## Gestió de processos automatitzats

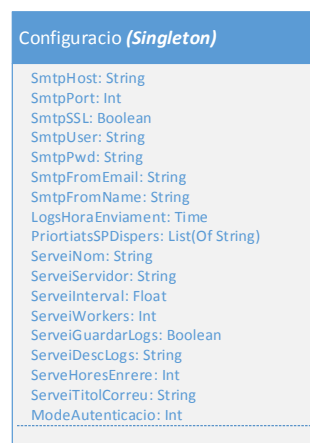
Caldrà que l'usuari pugui gestionar un procés en tota la seva complexitat, podent modificar el patró d'execució, activar-lo o desactivar-lo i executar-lo sota demanda. D'altra banda també caldrà poder incloure i excloure el procés en la llista de processos vigents en la planificació global. Finalment s'haurà de poder visualitzar el registre de processos, podent-lo netejar i filtrar a través d'un interval de dates a escollir.



### 6.3.2 Diagrama asociativu de classes

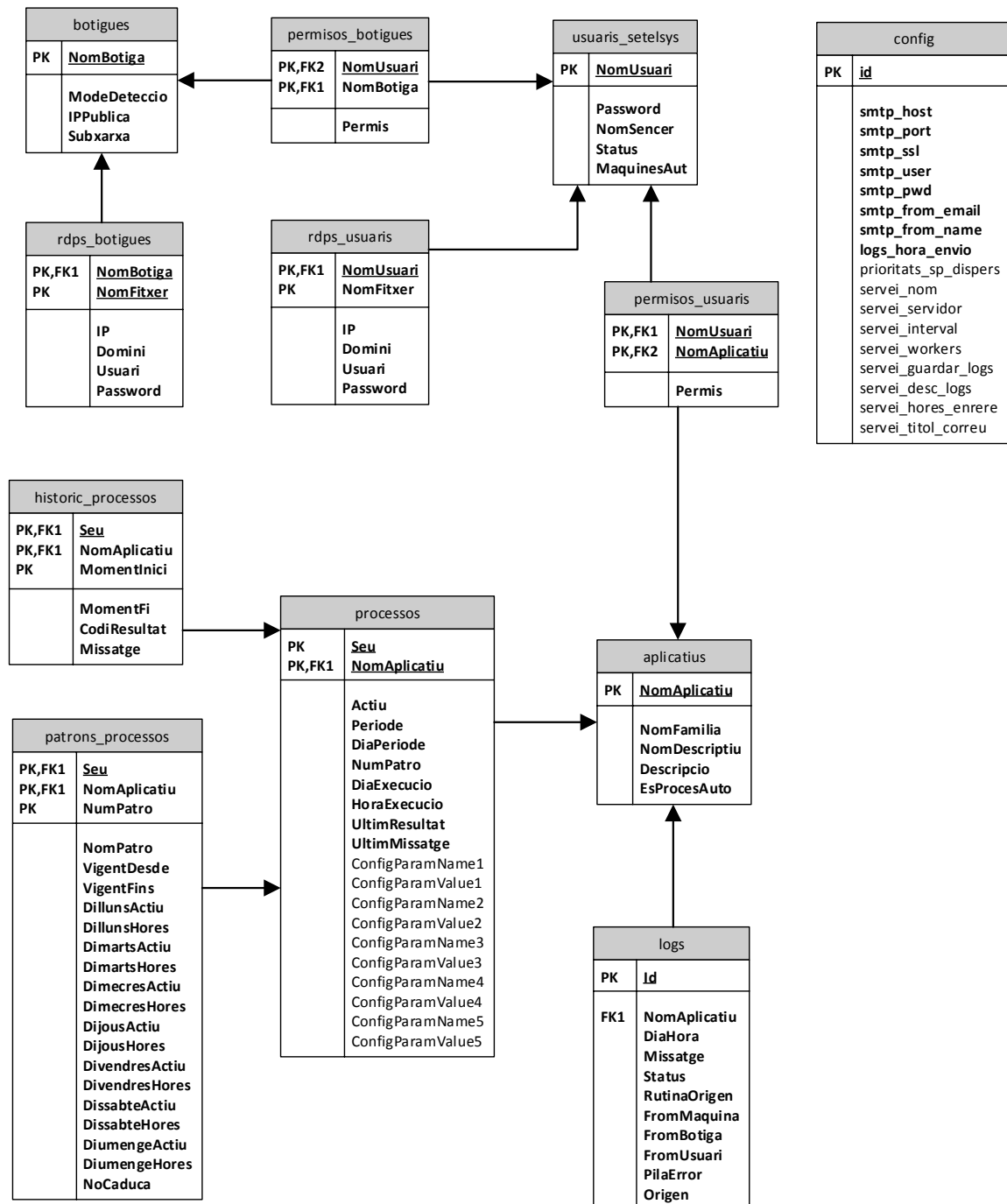


Existirà també una classe aïllada del diagrama, de tipus **Singleton** encarregada d'emmagatzemar paràmetres de configuració del sistema, anomenada **Configuració**:



## 6.4 Disseny de la base de dades de SetelSys

Mostrem tot seguit el model relacional resultant de l'especificació anterior, del qual tot seguit en mostrarem una explicació dels diferents grups de taules creats i la seva utilitat i funció com a membres del model de dades:



### *Gestió d'usuaris*

A través de la taula “*usuaris\_setelsys*” gestionem els diferents usuaris presents en el sistema, amb el seu nom, contrasenya i nom d'usuaris.

### *Gestió dels permisos dels usuaris*

Les màquines autoritzades decidim d'implementar-les com un atribut de tipus llista dins la taula d'usuaris. En canvi els permisos referents als aplicatius i les botigues, es guarden respectivament a “*permisos\_usuaris*” que formalitza l'implementació d'una relació N:N entre els usuaris els aplicatius, i a “*permisos\_botigues*” que a la seva part formalitza la relació N:N entre els usuaris i les botigues. També podem veure la relació existent entre la taula “*rdps\_usuaris*” amb la taula d'usuaris, que assigna els diferents *rdps* als corresponents usuaris.

### *Assignació de RDPs:*

Hem subdividit l'entitat RDP en dos taules “*rdps\_usuaris*” i “*rdps\_botigues*” que es relacionen amb usuaris i botigues respectivament. Hem detectat que ens certs casos es possible que en un futur es requereixi una especialització diferent per cadascuna, deixant així la porta oberta futures possibles modificacions o ampliacions de forma fàcil i trencant possibles dependències.

### *Botigues:*

Dins la taula “*botigues*” emmagatzemem les diferents botigues disponibles al sistema, juntament amb el seu nom identificador i el mètode que cal utilitzar per l'auto detecció de l'origen (utilitzant els atributs *IPPublica* i *Subxarxa*).

### *Aplicatius i processos:*

Dins la taula “*aplicatius*” s'emmagatzemen els diferents tipus d'aplicacions existents al sistema, així com la seva categorització (*NomFamilia*) i si es pot executar de forma automatitzada. En el cas de poder-se automatitzar es relaciona amb la taula “*processos*” on s'emmagatzema la informació del patró d'execució juntament amb la relació existent amb la taula “*patrons\_processos*” així com els paràmetres de configuració. Ens hem adaptat al màxim a la realitat de les necessitats per evitar costos de taules innecessàries, limitant el número de paràmetres a configurar en 5, amb la qual cosa podem definir-los dins la taula “*processos*” i no calent realitzar una nova taula i noves relacions.

### *Registre d'errors i històric de processos*

En la taula “*logs*”, guardem la informació de qualsevol error registrat al sistema amb tota la informació que el defineix així com la seva traçabilitat en la pila de l'error. D'altra banda a la taula “*històric\_processos*” guardem tota la informació referent a les diferents execucions realitzades dels diferents processos automatitzats vigents al sistema.

## 7. Desenvolupament

### 7.1 Metodologia utilitzada per al desenvolupament

Analitzem en aquest apartat diferents elements d'interès global en el procés de desenvolupament pel que fa als mètodes i tècniques utilitzades:

- [Eines utilitzades.](#)
- [Estructuració física dels arxius en l'entorn de desenvolupament](#)
- [Automatitzacions en el procés de compilació i desplegament](#)
- [L'ordre i cronologia aplicades.](#)
- [Tècniques de documentació utilitzades.](#)
- [Premisses i consideracions que s'han tingut en compte durant el desenvolupament.](#)

#### 7.1.1 Eines utilitzades

Per al desenvolupament de SetelSys ha fet falta l'ús de diferents eines destinades al procés de producció i de test de l'entorn:

- [Microsoft Visual Studio .NET 2008-2012](#): S'ha utilitzat l'entorn de programació en .NET de Microsoft per l'elaboració de tots els components.
- [Microsoft SQL Server 2008](#): Pel desplegament de la base de dades del sistema i la comunicació amb les bases de dades existents.
- [Microsoft SQL Studio Management 2008](#): Per l'edició de les bases de dades, taules, consultes, Transact SQL i en general tota l'operativa de gestió de dades a baix nivell.
- [Microsoft Visio 2010](#): Per la creació de diagrames i documentació necessària.
- [Microsoft Silverlight SDK v4](#): Per la creació de les interfícies destinades a la interacció amb l'usuari.
- [Microsoft Blend Expression for Visual Studio](#): Per la personalització de les interfícies Silverlight anotades en el punt anterior.
- [Microsoft IIS](#): Servidor web de Microsoft per l'allotjament de l'aplicació en producció.
- [Llibreries](#): S'han utilitzat les llibreries incloses dins el .NET Framework, així com algunes llibreries auxiliars externes que documentem en un altre punt d'aquesta secció.

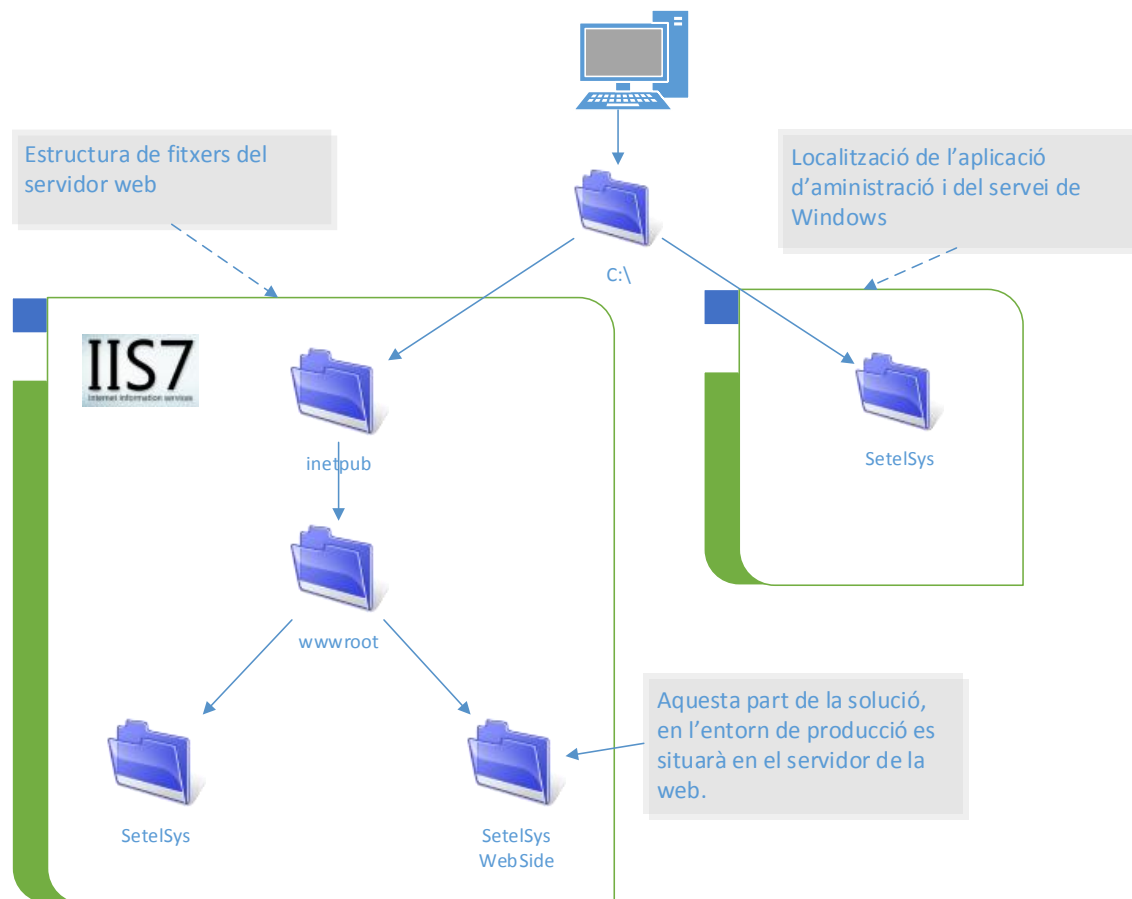
### 7.1.2 Estructuració física dels arxius de l'aplicació

Per tal de poder verificar el correcte desenvolupament de tot el sistema, es crea una estructura idèntica que la que existirà en el servidor que donarà servei a la versió en producció. En concret existiran dos focus en l'organització dels fitxers:

- Carpetes que contenen l'aplicació web **SetelSys** i el servei web **Maiden**
- Carpeta que conté l'eina administrativa del **SetelSysRunner** i el servei de Windows **SetelSysWinService**.

En totes elles apareixerà una referencia a la llibreria encapsulada del **SetelSysCore** per accedir a tota la funcionalitat disponible del sistema.

Veiem en el següent diagrama, l'estructura de fitxers tal i com s'ha organitzat en el servidor de desenvolupament i verificació, que correspon amb la que s'instal·larà en **producció**:



Dins de la carpeta "SetelSys" de l'estructura de fitxers del IIS, existeixen 3 carpetes que considerem el motor de l'aplicació web del SetelSys:



**Carpeta “bin”:**

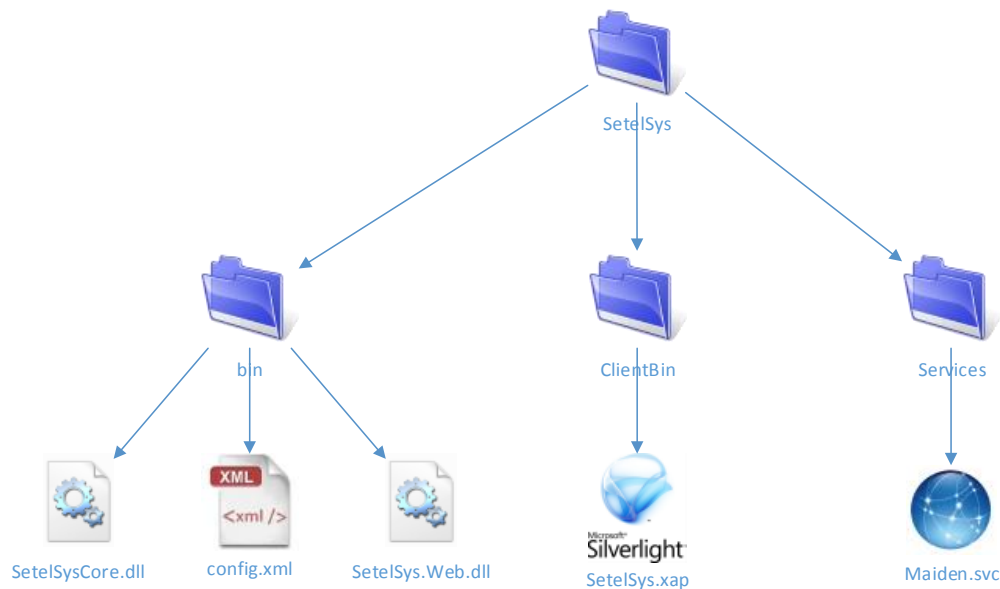
- **SetelSysCore.dll**: És la llibreria que conté tota la funcionalitat del nucli i necessària per la comunicació del servei **Maiden** amb el motor de l'aplicació.
- **SetelSys.Web.dll**: Representant del component en ASP.NET encarregat de carregar l'arxiu del SilverLight.
- **Config.xml**: Arxiu de configuració amb les dades de connexió de l'aplicació a les bases de dades per a instanciar correctament les classes **SQLServer**.

**Carpeta “ClientBin”:**

- **SetelSys.xap**: És l'arxiu resultant de la compilació del component dissenyat amb Silverlight i que s'encapsula sota aquesta extensió .XAP.

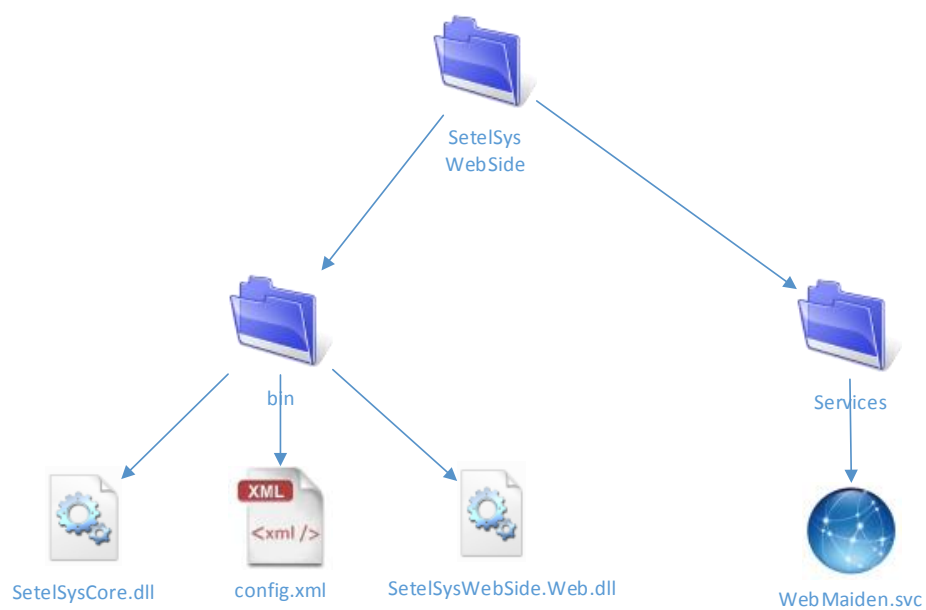
**Carpeta “Services”:**

- **Maiden.svc**: Amb la definició del servei web **Maiden**, que està implementat dins la llibreria “SetelSys.Web.dll” resident en la carpeta “bin”.

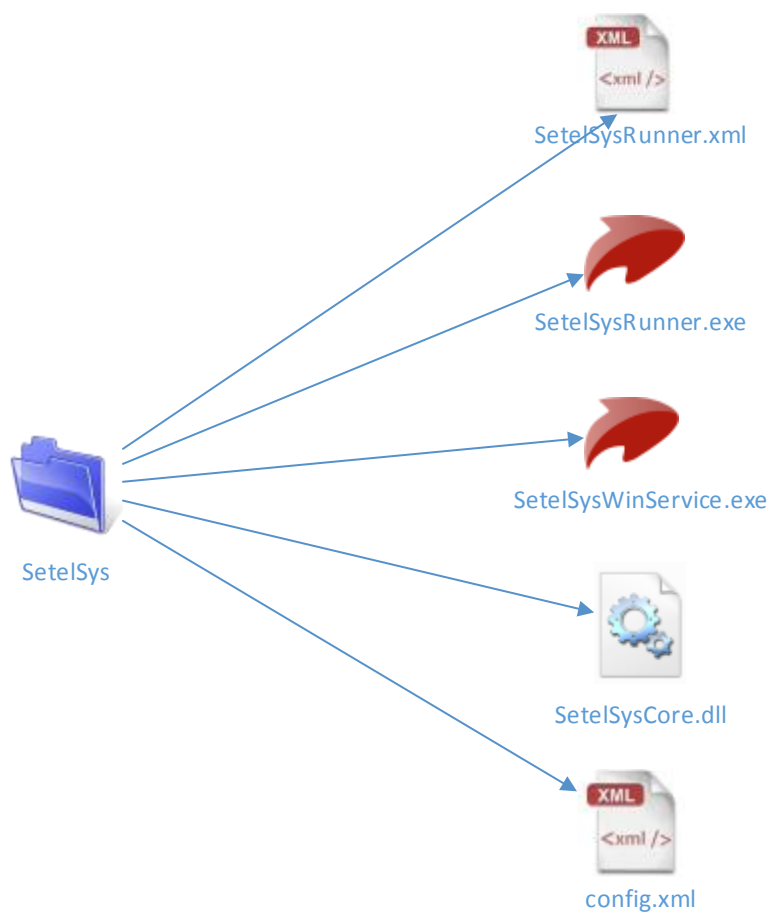


Si analitzem el contingut de la carpeta “**SetelSysWebSide**”, veurem que l'estructura és molt similar a l'anterior “**SetelSys**”.

La diferència resideix en que aquest component no implementa cap interfície SilverLight, sinó que simplement implementa el servei web necessari en el servidor de la pàgina web comercial de l'empresa per poder comunicar el nostre sistema amb ella i adquirir o bolcar les dades i informació necessària. Cal tenir en compte doncs, que aquest component, tot i que en l'entorn de desenvolupament el tenim dins la mateixa estructura de carpetes del nostre servidor IIS, en l'entorn de producció residirà en dit servidor de la web comercial.



Finalment, si analitzem el contingut de la carpeta “SetelSys” existent en l’arrel del nostre disc, podem veure com existeixen els següents fitxers a dins:



Con en el cas anterior, tant l'aplicació d'administració "SetelSysRunner.exe", com el servei de Windows "SetelSysWinService.exe" requereixen d'accés a la llibreria contenidora del nucli principal del sistema "SetelSysCore.dll". Tanmateix són necessaris els arxius de configuració "config.xml" i "SetelSysRunner.xml", aquest últim amb la configuració concreta de l'aplicatiu d'administració.

### 7.1.3 Automatització de l'entorn de desenvolupament i pre-producció

Per agilitzar el màxim el procés de desenvolupament, s'ha configurat l'equip del desenvolupador perquè el procés d'assemblatge de la solució estigui dotat dels esdeveniments i configuracions necessàries per actualitzar automàticament l'entorn de proves.

Degut al funcionament de l'IIS i les seves aplicacions, cada cop que es compila la solució fa necessari actualitzar els fitxers de diferents ubicacions. Per tal d'evitar que cada procés de compilació impliqui copiar i actualitzar fitxers de diferents ubicacions i per tant, perdre una estona en cada compilació per verificar-ne els resultats desitjats, hem configurat el procés de compilació perquè cada cop que s'executa, desi els arxius en les ubicacions necessàries per a deixar actualitzada l'aplicació.

Per tant, al finalitzar qualsevol compilació de la solució completa, queden disponibles al moment els accessos a tots els components de l'entorn, simulant al 100% l'entorn de producció final i podent detectar també anomalies per futures implantacions.

A l'efectuar una compilació de tota la solució, podem veure en el text següent extret de la sortida del compilador, com es mouen a la ruta de l'entorn corresponent, cada un dels components del sistema:

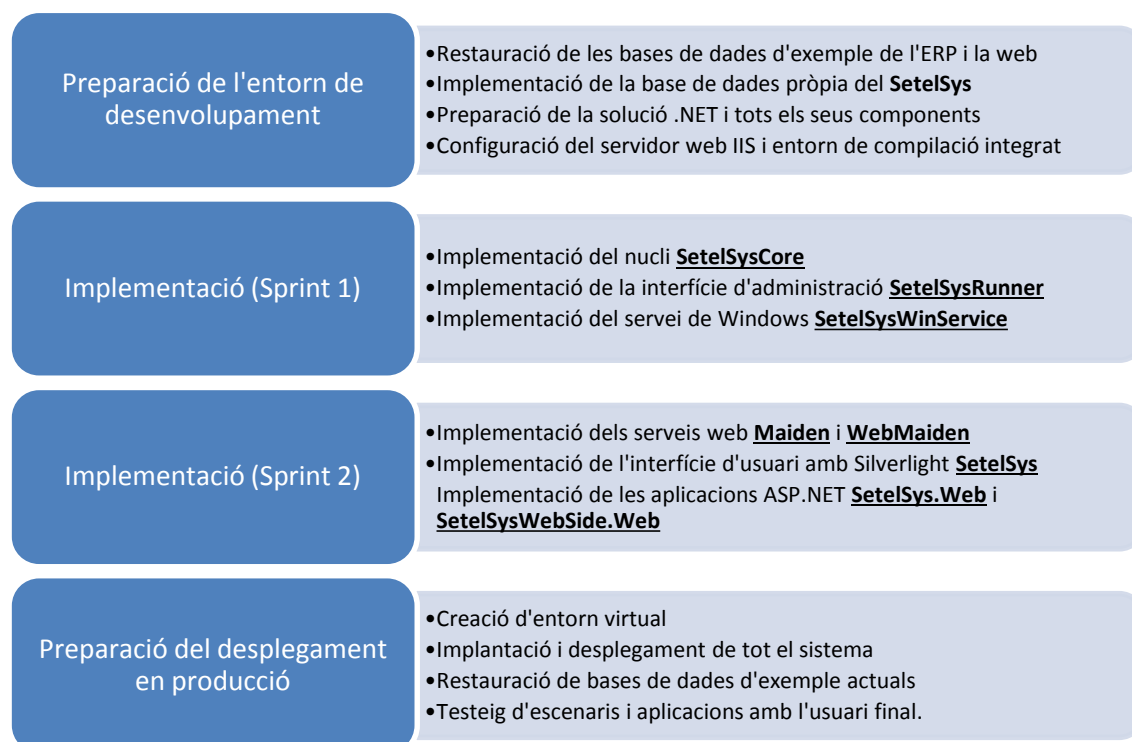
```
----- Rebuild All started: Project: SetelSysCore, Configuration: Debug Any CPU -----
SetelSysCore -> C:\SetelSys\SetelSysCore.dll
----- Rebuild All started: Project: SetelSys.Web, Configuration: Debug Any CPU -----
SetelSys.Web -> C:\inetpub\wwwroot\SetelSys\bin\SetelSys.Web.dll
----- Rebuild All started: Project: SetelSysWebSide.Web, Configuration: Debug Any CPU
SetelSysWebSide.Web -> C:\inetpub\wwwroot\SetelSysWebSide\bin\SetelSysWebSide.Web.dll
----- Rebuild All started: Project: SetelSysRunner, Configuration: Debug Any CPU
SetelSysRunner -> C:\SetelSys\SetelSysRunner.exe
----- Rebuild All started: Project: SetelSys, Configuration: Debug Any CPU
SetelSys -> C:\Projectes\SetelSys\SetelSys\Bin\Debug\SetelSys.dll
Begin Xap packaging
Creating file SetelSys.xap
Xap packaging completed successfully
SetelSys Xap file -> C:\inetpub\wwwroot\SetelSys\ClientBin
----- Rebuild All started: Project: SetelSysWinService, Configuration: Debug: Any CPU
SetelSysWinService -> C:\SetelSys\SetelSysWinService.exe
===== Rebuild All: 6 succeeded, 0 failed, 0 skipped =====
```

### 7.1.4 Ordre i cronologia en el desenvolupament

L'ordre seguit en el desenvolupament ha estat estructurat en 3 grans fases:

- Preparació de l'entorn de desenvolupament.
- Desenvolupament paral·lel d'interfícies i funcionalitats
- Preparació de l'arrencada en l'entorn de producció.

Veiem en el següent gràfic aquest desenvolupament per fases i a grans trets els elements principals de cada fase:



### 7.1.5 Modificació dels noms dels aplicatius

Per tal de facilitar una interpretació coherent i més intuïtiva dels noms que fins ara es tenien assignats als diferents aplicatius, alguns d'ells s'han modificat per evitar confusions i ambigüitats. Per exemple:

- “Pedidos” → Passa a anomenar-se “Pedidos entre tiendas”
- “Pedidos Todo Tiendas” → Passa a anomenar-se “Servir pedidos”
- “Articulos” → Passa a anomenar-se “Crear artículos en ERP”
- ...

### 7.1.6 Documentació de la codificació

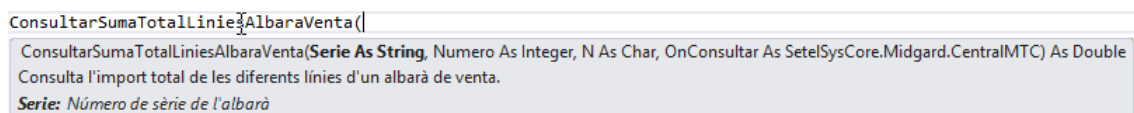
Durant el procés s'ha documentat al màxim possible qualsevol classe, mètode, funció, variable,... que hem vist que podia resultar poc clara per posteriors anàlisis del codi, ja sigui per nosaltres mateixos o per futurs desenvolupadors que hagin de manipular-lo.

Hem documentat els diferents punts del codi que creiem interessants i les capçaleres de funcions i mètodes també utilitzant el sistema propi del que disposa Visual Studio. Vegem-ne un exemple:

```
''' <summary>
''' Consulta l'import total de les diferents línies d'un albarà de venta.
''' </summary>
''' <param name="Serie">Número de sèrie de l'albarà</param>
''' <param name="Numero">Número de l'albarà</param>
''' <param name="N">Caràcter de control de l'albara, pot ser B o N</param>
''' <param name="OnConsultar">Seu a on consultar la informació</param>
''' <returns>Retorna el valor acumulat del total de les línies</returns>
''' <remarks></remarks>
Private Function ConsultarSumaTotalLiniesAlbaraVenta(Serie As String, Numero As
Integer, N As Char, OnConsultar As CentralMTC) As Double
```

D'aquesta forma quan intentem cridar la funció, i gràcies a la tecnologia **Intellisense**

<sup>13</sup>integrada dins el Microsoft Visual Studio, sempre ens apareix un missatge explicatiu utilitzant la definició emplenada durant la codificació de dita funció:

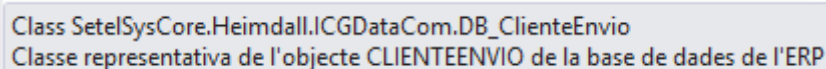


```
ConsultarSumaTotalLiniesAlbaraVenta(
ConsultarSumaTotalLiniesAlbaraVenta(Serie As String, Numero As Integer, N As Char, OnConsultar As SetelSysCore.Midgard.CentralMTC) As Double
Consulta l'import total de les diferents línies d'un albarà de venta.
Serie: Número de sèrie de l'albarà
```

En el cas de les classes, aquelles que hem cregut oportú, també s'han documentat utilitzant el mateix sistema, tant la classe en si, com els seus atributs, com veiem en el següent exemple:

```
'''<summary>Classe representativa de l'objecte CLIENTEENVIO de l'ERP</summary>
<Serializable()>
Public Class DB_ClienteEnvio
    ''' <summary>Codi del client</summary>
    Public Property CODCLIENTE As Integer = -1
    ...
End Class
```

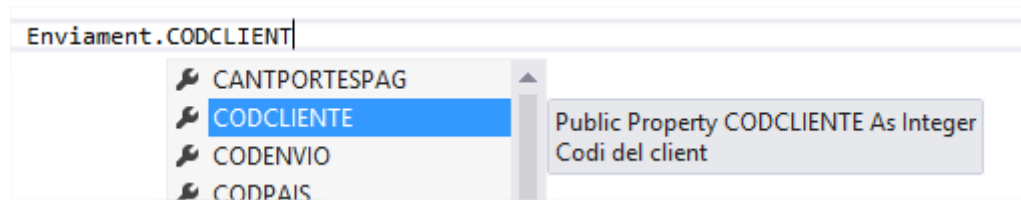
Obtenint un resultat com el següent a l'intentar instanciar un objecte d'aquesta classe:



```
Class SetelSysCore.Heimdall.ICGDataCom.DB_ClienteEnvio
Classe representativa de l'objecte CLIENTEENVIO de la base de dades de l'ERP
```

<sup>13</sup> Utilització de Intellisense a través de Visual Studio – [Microsoft MSDN](#)

De la mateixa manera a l'intentar accedir a un camp prèviament documentat com en l'exemple, també obtenim la informació del sumari.



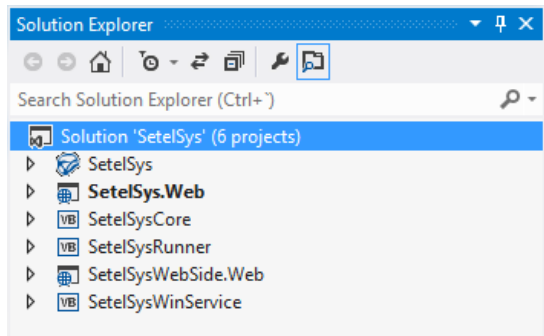
### 7.1.7 Premisses i consideracions presents durant el desenvolupament

Durant tot el cicle de desenvolupament hem seguit una normativa interna per assegurar el compliment de certes condicions que considerem bàsiques i primordials:

- Evitar la **redundància** o duplicitat de codi.
- **Reaprofitar** al màxim el codi, de manera que quan es detecta una funcionalitat que requereix d'un ús repetitiu s'encapsularà dins una nova classe, funció o mètode (depenent de la seva naturalesa semàntica) i així poder-la reutilitzar.
- Codificar **ordenadament** totes les funcionalitats en la classe que pertoca i en el cas de detectar-se la necessitat d'una nova classe, dissenyar-la i adaptar-la al sistema per fer el màxim d'adaptable possible la solució.
- **Parametritzar** al màxim les funcions i procediments, per tal de que futures funcionalitats o canvis esperats pel client es puguin transmetre retocant el mínim possible l'estructura de codi. Evidentment hem cercat un equilibri entre el cost de desenvolupament i l'obertura de les funcionalitats.
- Implementar totes les funcionalitats amb una [verificació bàsica](#) per obtenir una robustesa esperada en el resultat.
- **Documentar** al màxim possible totes les codificacions, així com les definicions de classes, mètodes i funcions, en aquells punts on es consideri necessari.
- Aplicar la [nomenclatura](#) explicada en l'apartat corresponent d'aquest document.

## 7.2 Estructura de la solució .NET

Com apareix en la següent imatge, podem visualitzar l'estructura de la solució .NET amb els diferents components que la componen i que més endavant tractarem de forma individualitzada.



La tipologia de cada un dels components que conformen la solució és la següent:

- **SetelSys:** Aplicació Silverlight<sup>14</sup>, hostatjada i iniciada a través del component SetelSys.Web.
- **SetelSys.Web:** Aplicació ASP.NET que inicialitza i carrega l'arxiu .XAP<sup>15</sup> generat en la compilació de SetelSys, i a la vegada allotja el servei web Maiden, basat en WCF<sup>16</sup>, per la interacció del component SetelSys amb els objectes del sistema.
- **SetelSysCore:** Llibreria de Windows (.DLL) encarregada d'implementar tot el nucli neuràlgic del sistema, representant els objectes i capes necessàries per suplir tota la lògica de negoci necessària.
- **SetelSysRunner:** Aplicació basada en WinForms<sup>17</sup> per actuar com a interfície d'usuari d'administració i parametrització del sistema, utilitzant SetelSysCore com a element de connexió amb la lògica de negoci i la gestió dels objectes de l'aplicació del sistema principal.
- **SetelSysWebSide.Web:** Aplicació ASP.NET<sup>18</sup> encarregada d'hostatjar el servei web necessari en l'allotjament de la web comercial per poder accedir a les dades i objectes representatius de la web.
- **SetelSysWinService:** Servei de Windows (Windows Service<sup>19</sup>) que implementa el motor de gestió d'execució de processos automatitzats.

<sup>14</sup> Que és Silverlight? – [Microsoft.com](http://Microsoft.com)

<sup>15</sup> Definició sobre els arxius .XAP – [Wikipedia](http://Wikipedia)

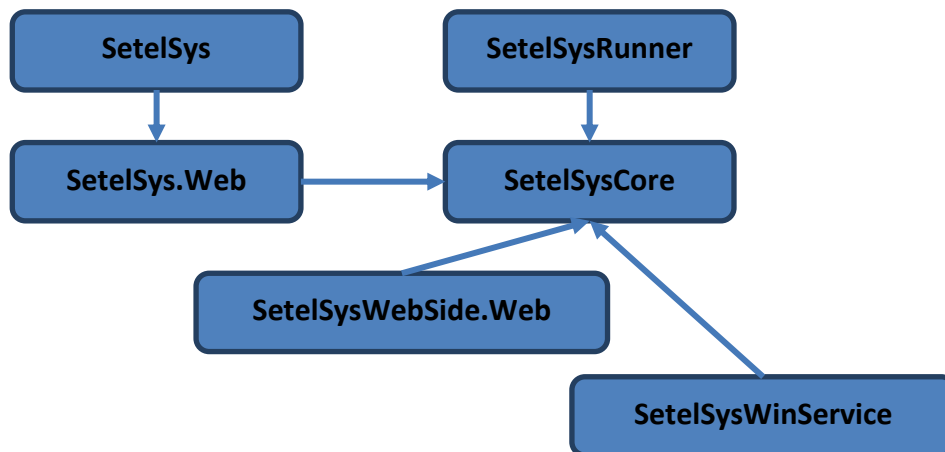
<sup>16</sup> Hosting and consuming WCF Services – [Microsoft MSDN](http://Microsoft MSDN)

<sup>17</sup> Windows Forms – [Microsoft MSDN](http://Microsoft MSDN)

<sup>18</sup> ASP.NET Web Forms – [ASP.Net](http://ASP.Net)

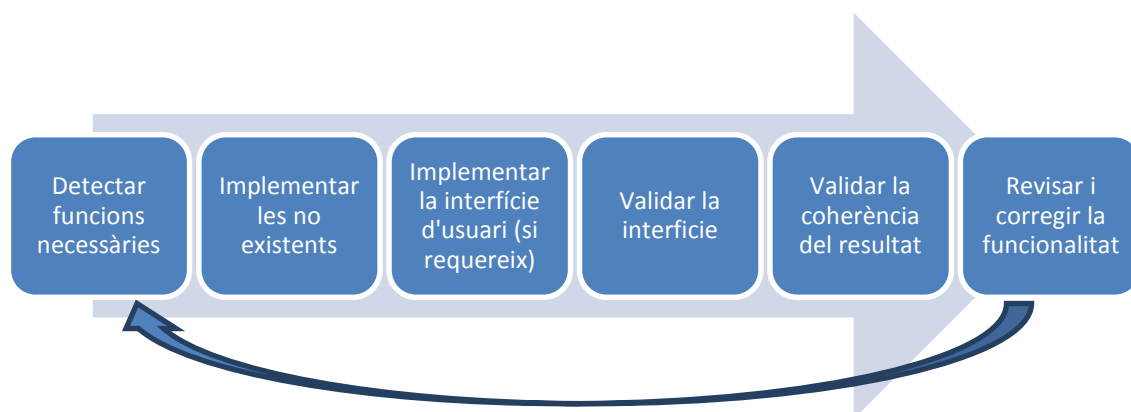
<sup>19</sup> Introduction to Windows Service Application – [Microsoft MSDN](http://Microsoft MSDN)

En el següent diagrama mostrem les referències existents en la solució, entre els diferents projectes dels components existents, i que en el fons responen a l'especificació de l'arquitectura de la solució:



### 7.3 Cicle de vida en la implementació de funcionalitats

Durant el desenvolupament de noves funcionalitats, s'ha seguit un procés encadenat per complir amb les premisses desitjades en el sistema resultant. En concret en tota nova funcionalitat implementada s'ha prosseguit seguint el següent diagrama representatiu:





## 7.4 Consideracions preses en el disseny de les interfícies i interaccions amb l'usuari.

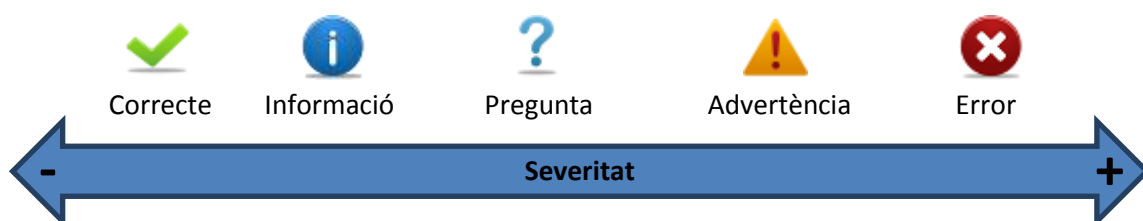
En la implementació de les interfícies d'usuari, ja sigui amb WinForms (SetelSysRunner) o Silverlight (SetelSys), s'han pres un seguit de consideracions pel que fa a les interaccions que es generen amb l'usuari. Les enumerem tot seguit:

### 7.4.1 Notificacions amb finestra emergent

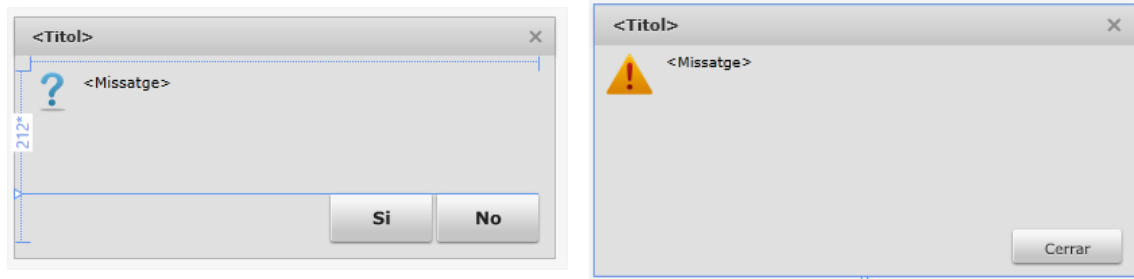
Hem classificat els diferents tipus de notificacions que es mostren durant l'execució d'algun procés o interacció amb la interfície per part de l'usuari, en tres nivells, que es podrien classificar de més a menys severitat:

- **Avisos informatius:** Sempre que es vulgui donar una informació sobre un resultat correcte, o informació extra sobre una funcionalitat o element de la interfície. El missatge associat serà purament informatiu.
- **Pregunta amb selecció requerida:** Quan calgui validar que l'usuari ha d'acceptar o cancel·lar algun procediment o simplement consultar una pregunta en la que els possibles resultats són únicament "Sí" o "No". El missatge associat serà una pregunta suficientment aclaridora sobre el seu motiu.
- **Avisos d'advertència:** Quan l'usuari ha de prendre alguna mesura a solucionar en la introducció de dades o la selecció d'elements, per exemple. El missatge associat indicarà que cal solucionar o completar, o simplement que cal tenir en compte.
- **Avisos d'error:** Quan s'ha produït un error no esperat en el sistema i cal notificar-ho a l'usuari. El missatge associat indicarà al màxim el motiu de l'error i l'origen per poder-ho notificar a l'administrador del sistema.

Sempre associarem les mateixes icones a cada tipus de notificació:



Degut a que aquests missatges són molt corrents, hem desenvolupat unes finestres ja parametritzades per mostrar la informació, podent escollir el tipus de notificació o pregunta que es vol efectuar a l'usuari, deixant així aquestes classes a disposició de tots els elements del projecte en si. En el següent exemple mostrem dos finestres per a la interacció de l'usuari pel que fa a una pregunta i una notificació en el projecte SetelSys de Silverlight:



Durant la creació d'una nova instància de les finestres es comunica a la constructora els paràmetres a mostrar en el títol i missatge.

```
Public Enum TipusMissatge As Integer
    OK = 1
    Info = 2
    Avis = 3
    Critic = 4
End Enum

Public Sub New(Titol As String, Missatge As String, Tipus As TipusMissatge)
    InitializeComponent()
    SetInfo(Titol, Missatge, Tipus)
End Sub
```

### 7.4.2 Mecanismes d'ajuda a l'usuari en formularis o pàgines

En diferents moments cal mostrar informació d'ajuda a l'usuari per interpretar el funcionament u objectiu d'algun dels elements de la interfície. Per fer-ho utilitzen diferents mecanismes depenent de la quantitat d'informació o detall que calgui donar a l'usuari:

- **Ajudes contextuais:** Activades a través del moviment del ratolí sobre les icones que situarem en els punts necessaris on cal mostrar l'ajuda.
- **Finestra d'ajuda estàndard:** Hem creat una classe destinada a mostrar una finestra amb el contingut d'ajuda associat en el moment de fer una nova instància d'aquesta classe, de manera que sempre es pugui utilitzar de forma genèrica en diferents punt del codi.
- **Finestra d'ajuda especialitzada:** En els casos que requereixen més complexitat, es crearà una classe especial per cada ajuda especialitzada, per incloure allà les imatges, textos i objectes necessaris per a instruir l'usuari.

En el següent exemple podem veure un exemple d'ajuda contextual (1) i d'ajuda especialitzada (2), ambdós casos en una de les aplicacions de la interfície en Silverlight:

The screenshot shows a web application interface. At the top, there is a navigation bar with a button labeled 'Ayuda / Leyenda' (marked with a question mark icon and labeled '2') and a 'Filtrar' button. Below this is a table with columns: E, T, P, Ref., Descripción, Colo., Talla, U., Alm. The first row of data is highlighted in yellow and contains the text 'RZ-FG030B PROTECCION PIE UNICO...'. A mouse cursor is hovering over the 'P' column of this row, which has triggered a contextual help message (marked '1') that says 'Hay pedido de compra al proveedor.'.

Below the table, a detailed help window titled 'Leyenda Información Pedidos Pendientes' is open. It explains the status indicators for pending orders:

- Estado del pedido:**
  - Disponibilidad Parcial (no todas las líneas pendientes del pedido tienen stock)
  - Disponibilidad Parcial Compartida (Alguna de las líneas del pedido no dispone de stock, pero el resto disponen de stock destinado a mas de un pedido, sin poder servirlos todos)
  - Disponibilidad Total (El pedido esta disponible al 100% en stock.)
  - Disponibilidad Total Compartida (El pedido esta disponible al 100% en stock, però hay otros pedidos que requieren de este stock, sin poder servirlos todos)
  - No Stock (No se dispone de stock en tienda para servir el pedido)
- Otros Indicadores:**
  - Pedido no procesado
  - No hay disponibilidad FUTURA.
  - Hay faltas para el material del pedido.
- T = Stock en transito hacia vuestro almacén**
  - Si aparece este icono en esta columna, significa que hay unidades en transito para completar este pedido.
- P = Existen unidades Pedidas a la Central, que estaran pedidas a su vez al proveedor.**
  - Si aparece este icono en esta columna, significa que hay unidades pedidas a la CENTRAL.
  - Si aparece este icono en esta columna, significa que hay unidades pedidas a PROVEEDOR.
- C = Consulta al almacén**
  - Si aparece este icono, significa que hay una consulta abierta con la central para determinar el estado del pedido. Haciendo click en el icono se abrirá la pantalla de consultas, si no existe consulta abierta, puedes crear una haciendo click en el boton de "Crear Consulta"
  - Existe una fecha prevista de entrega para algunos o varios artículos del pedido.
  - Se ha anotado una NUEVA fecha prevista de entrega para algunos o varios artículos del pedido.

A 'Salir' button is located at the bottom right of the help window.

### 7.4.3 Utilització sistemàtica de controls

Per tal de mostrar unes interfícies homogènies i que permetin a l'usuari identificar clarament la funcionalitat de les diferents parts de les interfícies, s'ha utilitzat sempre el mateix tipus de control associat a un objectiu gràfic o d'introducció de dades similar, permetent a l'usuari tenir una idea clara dels objectius i el funcionament de la interfície a mesura que s'incrementa el seu aprenentatge en el sistema. En concret hem assignat les diferents tasques als següents controls<sup>20</sup>:

#### Visualització de dades:

- **Graelles** (*DataGrid* i *DataGridView*): Les utilitzarem per mostrar un llistat de dades complexes, que impliquen més d'un atribut a visualitzar per cada objecte.
- **Llistes** (*ListBox* i *ListView*): Les utilitzarem per mostrar un llistat de dades simples, on només cal mostrar un atribut de l'objecte.
- **Distribució en pestanyes** (*TabControl*): Quan la informació a mostrar és superior al que cap de manera ordenada en l'espai disponible, dividirem l'espai en diferents pestanyes.
- **Desplegables** (*ComboBox*): Quan la informació que es vol mostrar és simple (un sol atribut) i el volum d'elements a seleccionar és petit, utilitzarem llistes desplegables.
- **Etiquetes** (*Label*): Per mostrar informació curta i identificativa de l'objectiu dels diferents controls.

#### Introducció i selecció de dades:

- **Quadres de text** (*TextBox* i *TextView*): Per a introducció de textos curts d'una sola línia.
- **Blocs de text** (*TextBox* i *TextBlock*): Per a la introducció de textos llargs que impliquen l'entrada de més d'una línia de text.
- **Desplegables** (*ComboBox*): Per a la selecció d'un element dels disponibles en un grup i així minimitzar possibles errades en la introducció errònia de dades.
- **Graelles** (*DataGrid*, *DataGridView*): Quan el volum d'objectes disponibles a seleccionar és gran, utilitzarem les graelles en comptes dels desplegables.
- **Calendaris** (*Calendar*, *DateTimePicker*): Per la selecció de dates utilitzarem una vista de calendari, amb l'opció de que l'usuari pugui fer una selecció de dades més acurada.
- **Selectors numèrics**: (*NumericUpDown*): L'utilitzarem quan el valor a introduir ha de ser obligatòriament numèric i així assegurar la correctesa de les dades.
- **Selectors** (*Slider*, *TrackBar*): Per la selecció de dades dins d'un interval.

#### Execució d'accions:

- **Botons** (*Button*): Per a la majoria de casos utilitzarem aquest control, i sempre que sigui possible ajudant la interpretació amb una imatge a part del text informatiu.

<sup>20</sup> VB.Net Basic Controls ([TutorialsPoint.com](http://TutorialsPoint.com)) – Silverlight Controls ([Microsoft MSDN](http://Microsoft MSDN))

- **Imatges** (*Image, PictureBox*): Per a certes ordres s'utilitzarà l'ús d'imatges que al ser premudes activaran l'acció desitjada. Es reserva aquest ús a casos que no tenen possible interpretació errònia i suficientment intuïtiva.

#### 7.4.4 Utilització sistemàtica d'imatges identificadores

Al llarg de les interfícies s'utilitzen controls amb imatges o imatges aïllades per a interactuar amb les possibles accions a realitzar per l'usuari. Justament per això s'ha utilitzat de manera identificadora un seguit d'imatges perquè es puguin associar sempre a un mateix concepte **semàntic**. En la següent taula donem una llista de les imatges estandarditzades pel que fa a la seva semàntica:

Imatge	Valor semàntic
	Afegir
	Material en transit, Enviament
	Alerta
	Assistent, Parametrització
	Configuració
	Avís
	Bloquejat, desbloquejat
	Guardar els canvis
	Error, eliminar
	Acceptar, resultat correcte
	Imprimir
	Cercar
	Refrescar
	Informació
	Magatzem, Caixes, Paquets
	Sortir del formulari
	Editar
	Error de connexió
	Endavant, Endarrere
	Descarregar
	Consulta, conversa
	Esborrar
	Etiquetar, Categoria
	Base de dades, dades
	Selecció d'una data, informació sobre dates
	Cancel·lar

## 7.5 Implementació de la base de dades principal de SetelSys

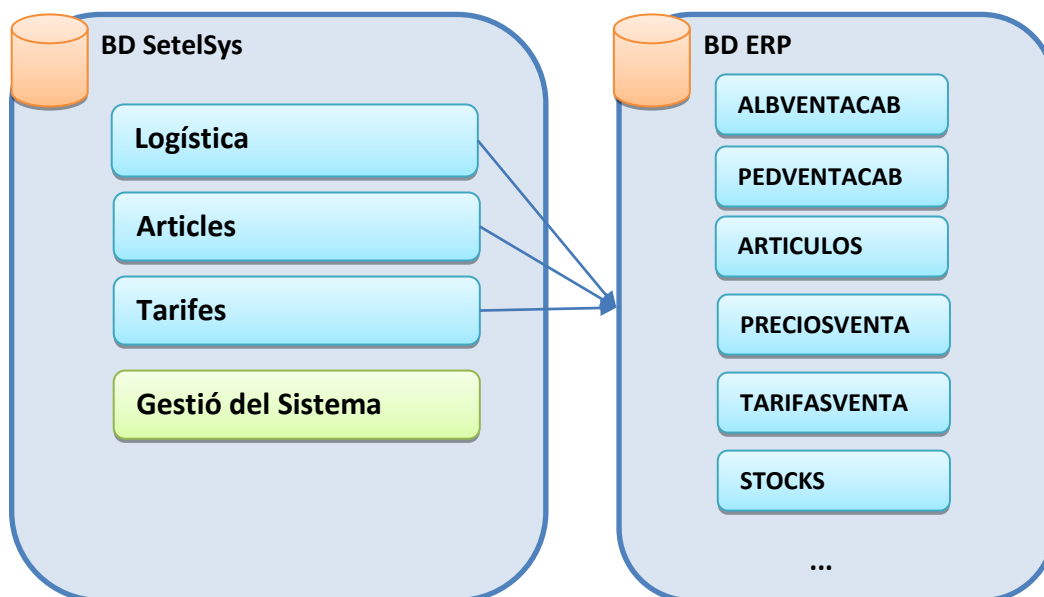
Primerament cal tenir en compte que la base de dades de SetelSys implementarà les entitats i relacions necessàries per:

- Gestionar la infraestructura del sistema SetelSys.
- Gestionar la interacció del sistema amb altre subsistemes: ERP + Web

Això genera una casuística especial, ja que si bé per les taules referents al sistema SetelSys podem gestionar-les totes dins la nostra base de dades a implementar, les taules relacionades amb l'ERP i la web comercial, **no poden plasmar les relacions existents al formar part de dos bases de dades independents lògica i físicament.**

Per aquest motiu, la base de dades de SetelSys implementa les entitats necessàries pel funcionament de les aplicacions dependents de l'ERP (les tasques a gestionar a la web comercial no requereixen de creació de taules auxiliars), sense existir una relació a nivell estructural de la base de dades, però si semàntic, entre les entitats.

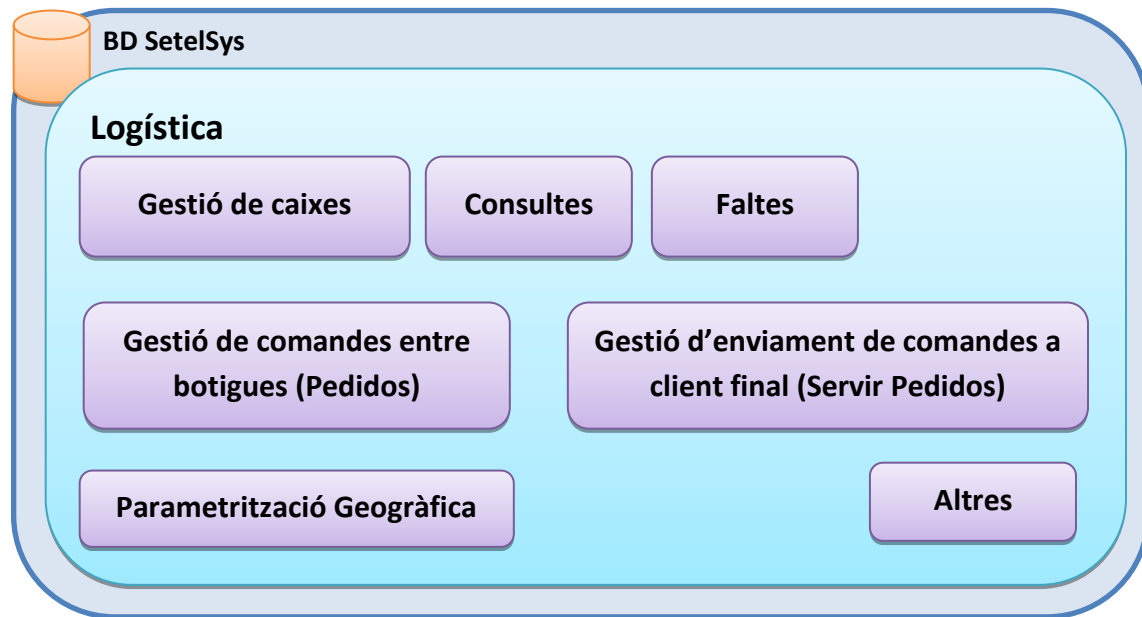
Mostrarem amb el següent diagrama els blocs de taules que componen la base de dades de SetelSys, tenint en compte que cada bloc és independent entre si (s'utilitza per finalitats molt diferents) i dependent amb el sistema ERP.



Les diferents taules existents en l'ERP i la seva definició a través d'un model relacional, ja s'ha tractat en l'anterior secció d'especificació. En aquesta secció ens centrarem en les diferents taules necessàries per a totes les funcionalitats desitjades pel client que amplien les existents pel sistema actual. Cal recordar en tot moment que SetelSys és un sistema satèl·lit i extern a l'ERP principal i per tant tota informació generada i que es desitgi emmagatzemar físicament en una base de dades, cal que sigui gestionada pel nostre sistema.



### 7.5.2 Taules per a l'ús d'aplicacions logístiques



Aquest bloc de taules és el més extens i complex, format per taules auxiliars necessàries per relacionar SetelSys amb l'aplicació ERP i dotar de la funcionalitat de consulta i gestió desitjada pel client. Cada un d'aquests sub blocs definits dins de Logística, només mantenen relació física a nivell de BD entre taules del mateix sub bloc, tot i que semànticament algunes estan relacionades amb l'ERP.

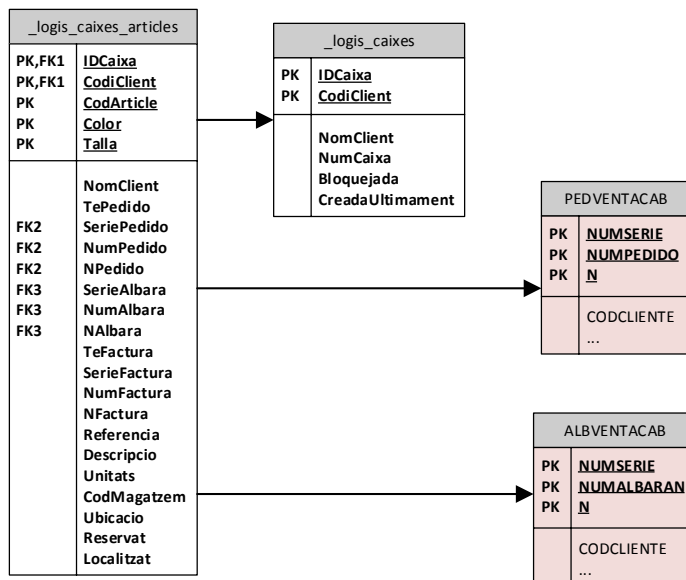
Desglossem cada un d'aquests sub blocs, mostrant el model relacional corresponent a les taules que el conformen.

Pintarem de color **especial** les taules que són de l'ERP i hi afegim una relació virtual, sent sempre conscients que aquesta relació és purament semàntica però no realment plasmada sobre la base de dades.



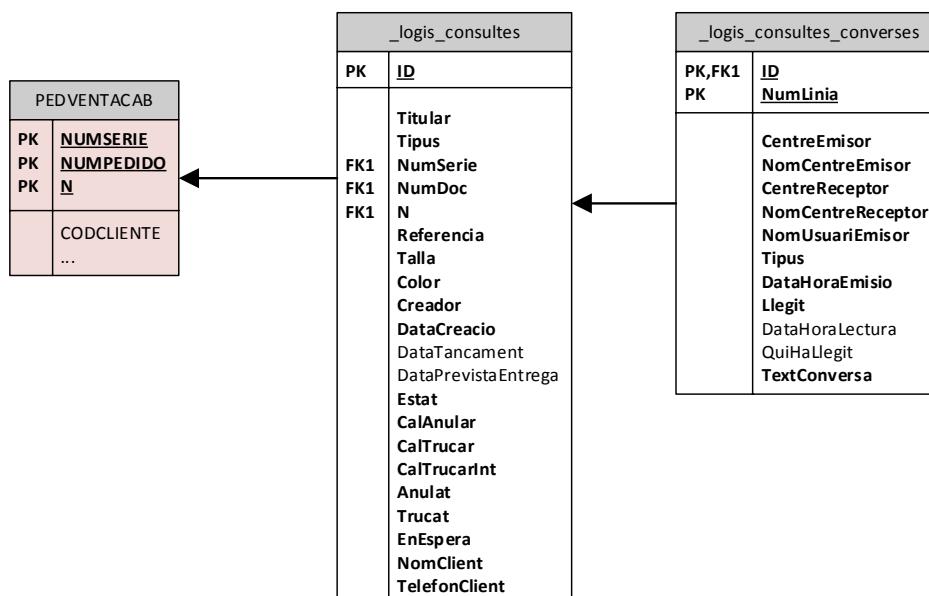
### Gestió de caixes:

Requerides per l'aplicació encarregada de generar caixes de tot el material que s'ha de preparar al magatzem central per l'enviament al client. S'agrupa tot el material en una sola caixa per cada client, en la qual s'afegeixen les diferents línies d'articles que cal reservar per aquest client, notificant en tot de la relació que té cada material amb la comanda pendent de servir al client i l'albarà que s'ha generat per entregar el material (com podem veure en la relació **virtual** existent entre els articles de la caixa i les taules PEDVENTACAB i ALBVENTACAB)



### Consultes:

Necessàries per poder emmagatzemar informació auxiliar referent a les consultes assignades a una comanda (relació **virtual** amb PEDVENTACAB), de forma que podem associar diferents consultes sobre una comanda d'un client, per un dels articles existents en ella. Dins d'una consulta es poden agregar les línies de conversa que es vulguin, anotant la informació de control que es desitja registrar en cada nou missatge.



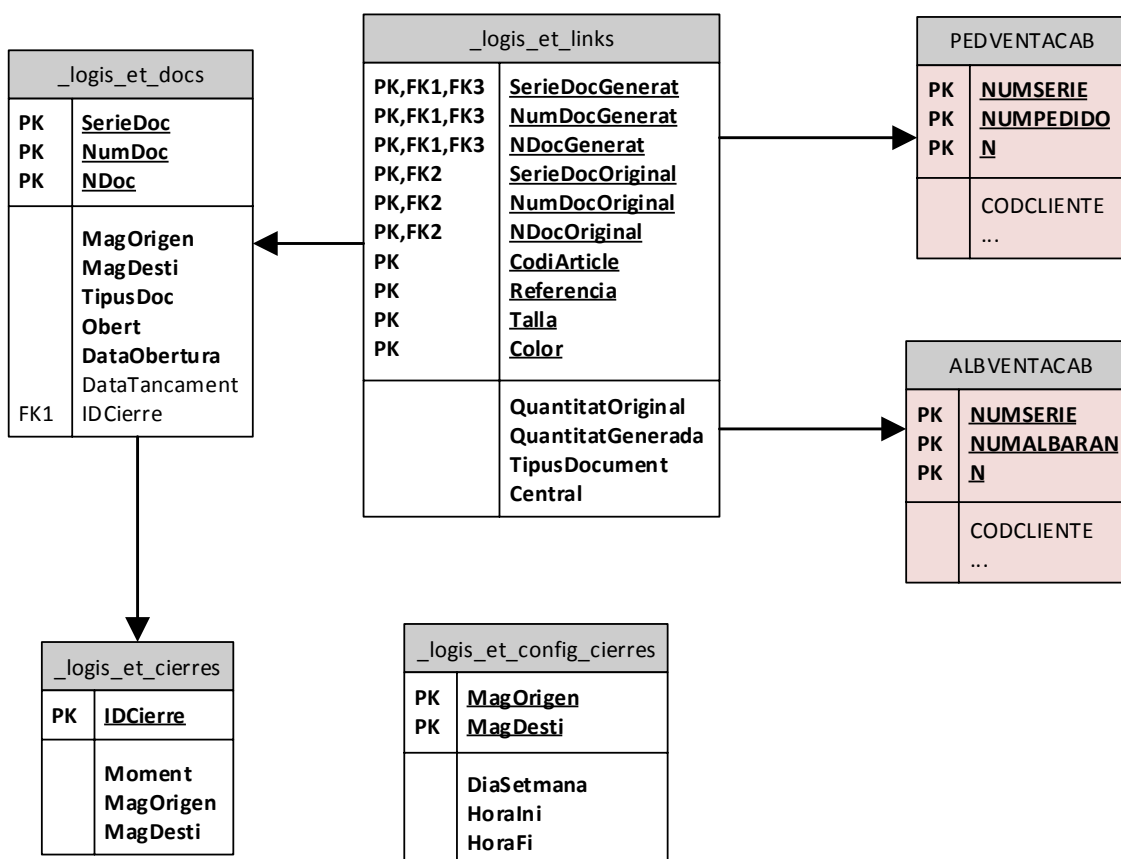
### *Gestió de les comandes entre botigues ("Pedidos"):*

Necessàries per generar les comandes entre botigues, per tal d'envia'ls-hi les unitats necessàries d'articles per servir al client des de la botiga.

S'utilitzen traspassos d'enviament setmanal (per reduir costos logístics), de tal manera que un cop a la setmana (configurable en el SetelSysRunner, utilitzant la taula \_logis\_et\_config\_cierres), s'obren traspassos entre les diferents botigues (\_logis\_et\_docs).

En aquests traspassos (ALBVENTACAB) s'afegeixen els articles necessaris per servir les comandes (PEDVENTACAB) i es registra la relació entre les diferents línies, la comanda a la que pertanyen i l'albarà o traspàs en el que s'han afegit (taula \_logis\_et\_links).

Finalment, un cop a la setmana es fa el tancament del document (taula \_logis\_et\_cierres), per tal de poder enviar definitivament el material entre les botigues i començar un nou cicle de repetició setmanal.



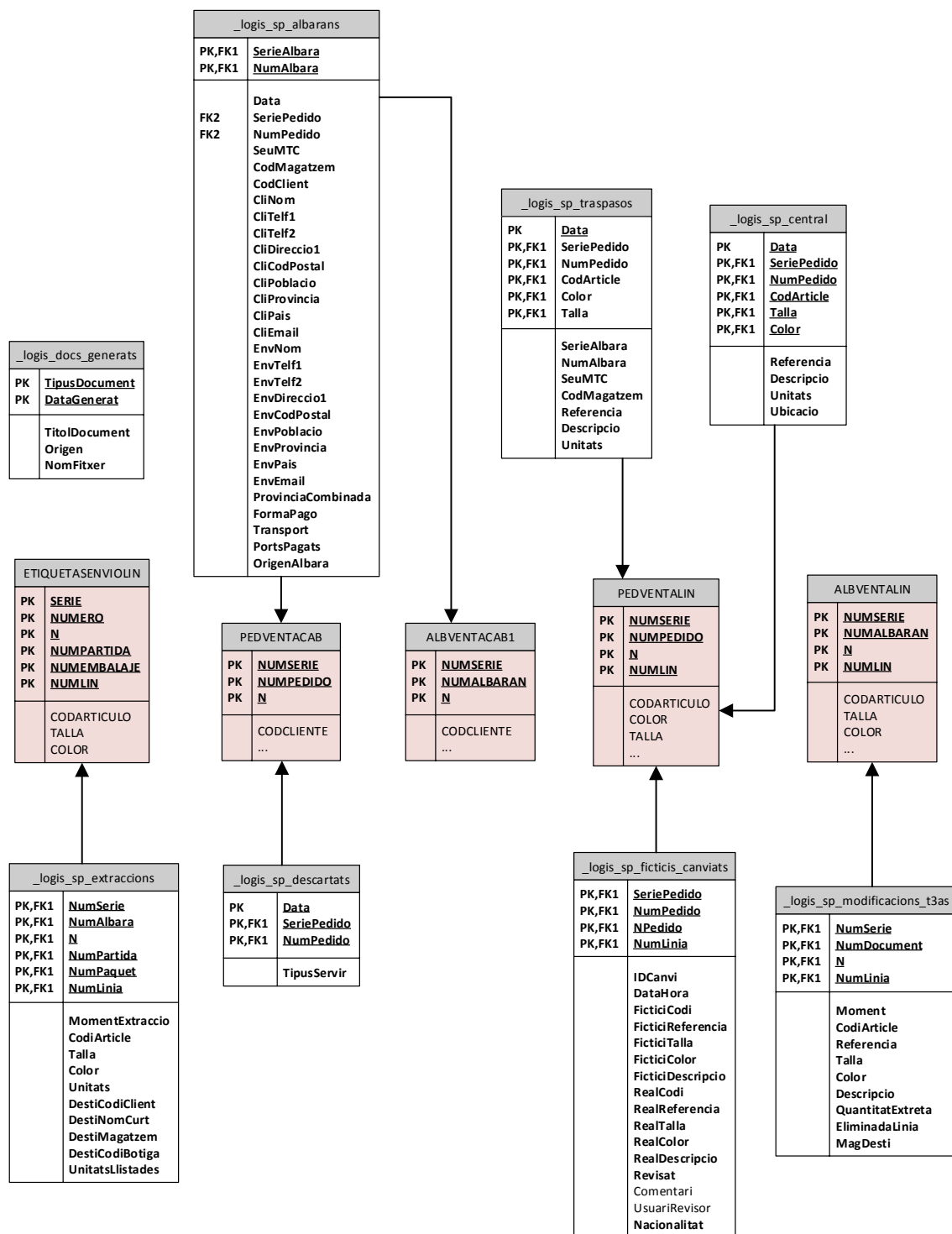
### *Gestió d'enviament de comandes a client final ("Servir Pedidos")*

En el procés de servir les comandes directes al domicili del client final, ha fet falta fer ús de bastantes taules auxiliars, convertint-se en el procés més complex de tots a nivell algorímic i computacional. Al servir les comandes, cal registrar tot un seguit d'informació necessària de plasmar en la base de dades:

- Dels albarans que es generen directament a l'ERP, cal guardar-ne la informació referent al client (per agilitat en còmputos i consultes posteriors) dins la taula `_logis_sp_albarans`, així com la relació a la comanda original del client i l'albarà en que s'ha servit (PEDVENTACAB i ALBVENTACAB).
- Cal guardar la informació de les comandes descartades per cada data d'execució, per motius de falta d'estoc i disponibilitat, a la taula `_logis_sp_descartats`.
- Dels traspassos que s'han hagut de generar des de les diferents botigues cap al magatzem central per poder-ne enviar el material cap al client, n'enregistrem la informació de l'article, data, etc. conjuntament amb la relació amb la comanda original del client (PEDVENTACAB) dins la taula `_logis_sp_traspassos`.
- De la mateixa manera cal guardar la informació de tot el material que ja està actualment disponible a la central per posteriorment poder generar els llistats corresponent per la preparació dels paquets, a la taula `_logis_sp_central`, guardant també la seva relació amb la comanda original del client (PEDVENTACAB)
- Durant el procés de servir les comandes es generen tot un seguit de documents de treball per distribuir les tasques als diferents elements organitzatius de l'empresa. Cada document que es genera es registra amb les dades del fitxer i la data en que es genera (taula `_logis_docs_generats`).

Pot ser necessari fer certes actuacions especials, que també cal registrar a la base de dades:

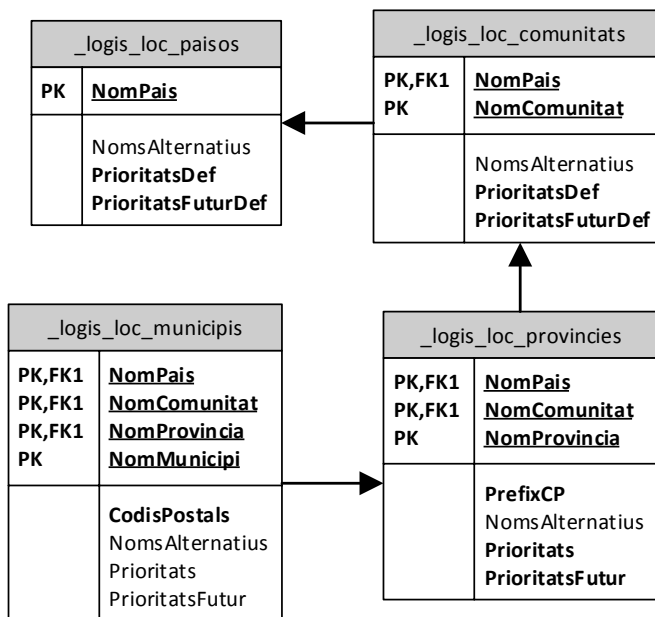
- Extreure material de traspassos utilitzats per l'estocatge addicional a les botigues, si aquest material pot fer falta per servir la comanda d'algun client (es prioritza l'enviament a client per sobre de traspassos d'estocatge). En cas d'extreure's del traspàs, s'anota la informació associada i el traspàs original d'on s'extreu (ALBVENTACAB) dins la taula `_logis_sp_modificacions_t3a`
- En el cas de que els anteriors traspassos d'estocatge esmentats, tenien assignats ja una *Packing List*, cal extreure'l de dins seu. Es guarden aquestes extraccions a `_logis_sp_extraccions`, conjuntament amb la seva relació amb la taula dels *Packing Lists* (ENVIOSLIN).
- Fer canvis d'articles considerats d'estoc fictici pel seu equivalent real, anotant les dades de l'article fictici, del real i la seva relació amb la comanda original del client (PEDVENTACAB), a través de la taula `_logis_sp_ficticis_canviats`.



### Parametrització geogràfica

Per tal de poder utilitzar en la configuració de les prioritats al servir les comandes al client final, necessitem registrar d'una manera suficientment flexible i organitzada les localitzacions geogràfiques. Per això hem definit la jerarquia PAISOS > COMUNITATS > PROVINCIES > MUNICIPIS tal i com mostren les 4 taules següents, amb l'afegit especial de que per cada entitat en registrem informació extra, com poden ser:

- Noms alternatius de les entitats (per fer més flexible el procés de reconeixement).
- Prioritats per defecte actuals i futures per cada entitat.
- Codis postals associats o prefix dels codis postals.



### Faltes:

Cal poder enregistrar a través de la següent taula auxiliar, les diferents faltes d'estoc que es donen d'alta en el sistema i que es tindran en compte en els càlculs d'estocs disponibles per totes les aplicacions afectades. De cada falta en registrem la informació corresponent a l'article, les unitats afectades, data, i informació de l'usuari que obre i tanca la falta, així com les descripcions associades.

_logis_et_faltes	
PK	IDFalta
	DataFalta
	Magatzem
	Codi
	Referencia
	Talla
	Color
	Quantitat
	Descripcio
	Revisada
	DescAnomalia
	DescSolucio
	UsuariObertura
	UsuariTancament
	Proces

### Altres:

Existeixen altres taules requerides per registrar elements necessaris per càlculs i operatives especials sobre l'ERP però de les quals el propi ERP no disposa de la capacitat de modificació i implantació en el seu model de dades, deixant la responsabilitat a SetelSys d'implementar dites necessitats:

- Configuracions de prioritats associades a articles pels càlculs de traspassos i comandes a proveïdor automatitzades (\_logis\_pedidos\_prioritats)
- Assignacions de proveïdors habituals per diferents marques d'articles, relacionat amb la taula MARCA de l'ERP, utilitzant la taula \_logis\_marques\_proveïdors.
- Agrupació de magatzems sota diferents tipologies (per ubicació física, per distribució de tarifes,...), utilitzant la taula \_logis\_grups\_magatzems.
- Emmagatzemar els telèfons llistats des de les botigues a través de la taula \_logis\_telfs\_impresos.

_logis_pedidos_prioritats	
PK	<u>CodMagatzem</u>
	CadenaPrioritat

_logis_marques_proveïdors	
PK	<u>CodiMarca</u>
	NomMarca CodiProveïdor NomProveïdor

_logis_grups_magatzems	
PK	<u>Nom</u>
PK	<u>Tipus</u>
	Magatzems

_logis_telfs_impresos	
PK	<u>Data</u>
PK	<u>Botiga</u>
	CodiClient Telefon

## 7.5.3 Taules per l'ús d'aplicacions relacionades amb els articles

### Creació massiva d'articles

Cal gestionar la creació de nous articles de forma massiva i més àgil que el sistema natiu de l'ERP, i cal guardar la informació dels articles creats per posteriors llistats i traspàs definitiu a l'ERP. Utilitzarem la taula \_arts\_nous per guardar tota la informació bàsica de l'article, mentre que les taules \_arts\_nous\_preus i \_arts\_nous\_minims emmagatzemen la informació dels diferents preus i mínims a guardar al crear el nou article.

### Configuració pel recàlcul de costos

Es necessari enregistrar tot un seguit de paràmetres per al recàlcul dels costos d'articles que efectua de forma deficient l'ERP, i que enregistrarem a \_arts\_config\_rec\_costos.

### Magatzems no autoritzats per traspassar-hi articles

Certs articles no es volen poder enviar en algun dels magatzems existents en els diferents aplicatius que s'executaran en el sistema. Per això enregistrem a la taula \_arts\_mags\_no\_autoritzats, l'article en qüestió (relacionat amb ARTICULOS) i els magatzems als quals no està autoritzat a enviar-se.

### Pesos

Es vol poder associar un pes predeterminat a les diferents combinacions possibles de departaments, seccions, famílies i subfamílies de l'ERP, a través de la taula \_arts\_pesos, per

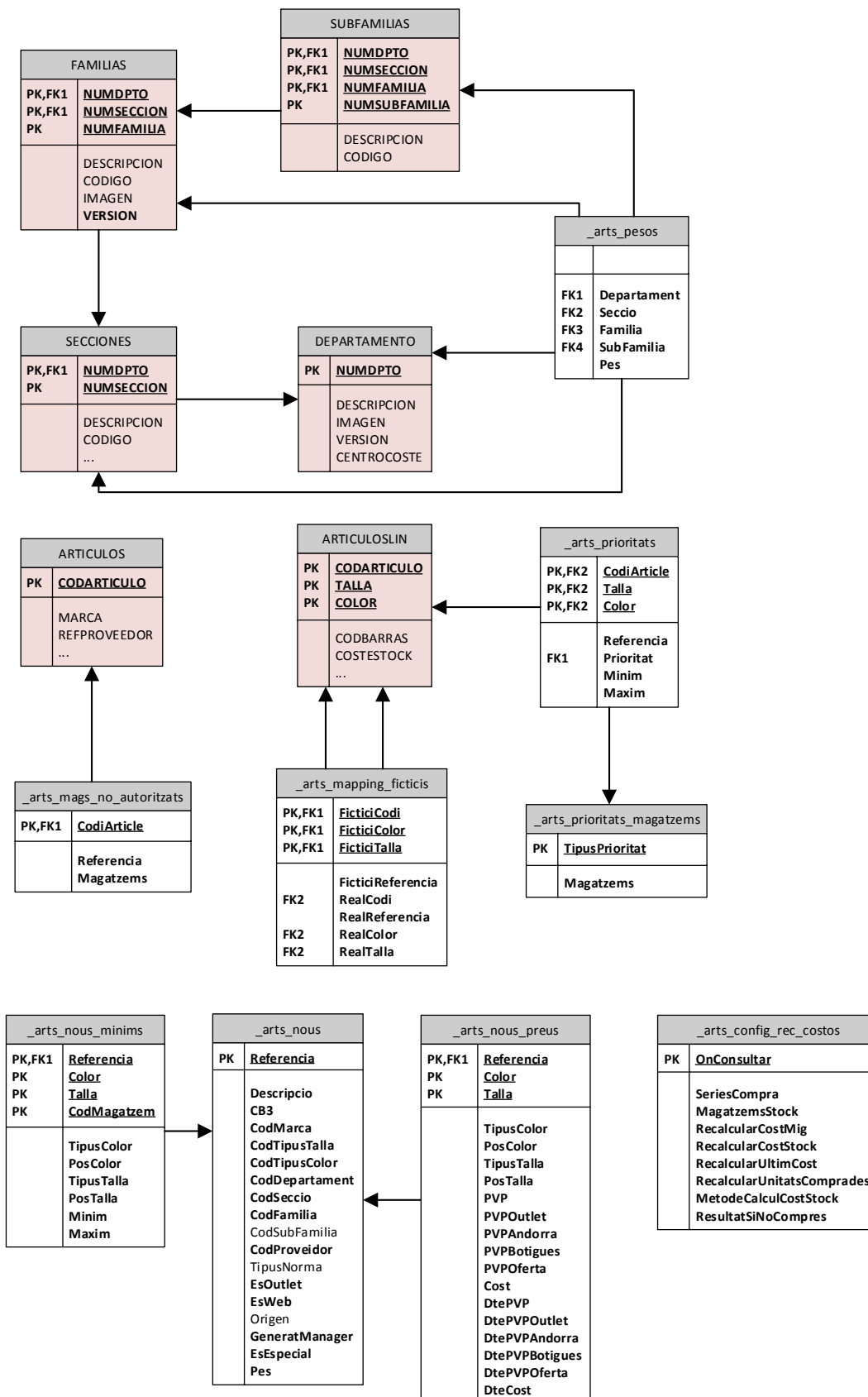
poder facilitar la informació corresponent al gestionar els enviaments i acotar el cost de transport.

### ***Prioritats***

Cal registrar les prioritats d'enviament en les propostes automatitzades de traspassos de balanceig i comandes, registrant en la taula \_arts\_prioritats, la prioritat assignada a cada article (relacionat amb ARTICULOSLIN) i que serà una de les existents en la taula \_arts\_prioritats\_magatzems, on es defineix quins magatzems pertoquen a cada prioritat.

### ***Mapeig d'articles ficticis i reals***

Cal poder enregistrar una relació entre els articles considerats ficticis i els reals per tal de poder facilitar l'assignació automàtica posterior al servir les comandes al client final. S'enregistra a la taula \_arts\_mapping\_ficticis, on tant l'article real com el fictici es relacionen amb ARTICULOSLIN.

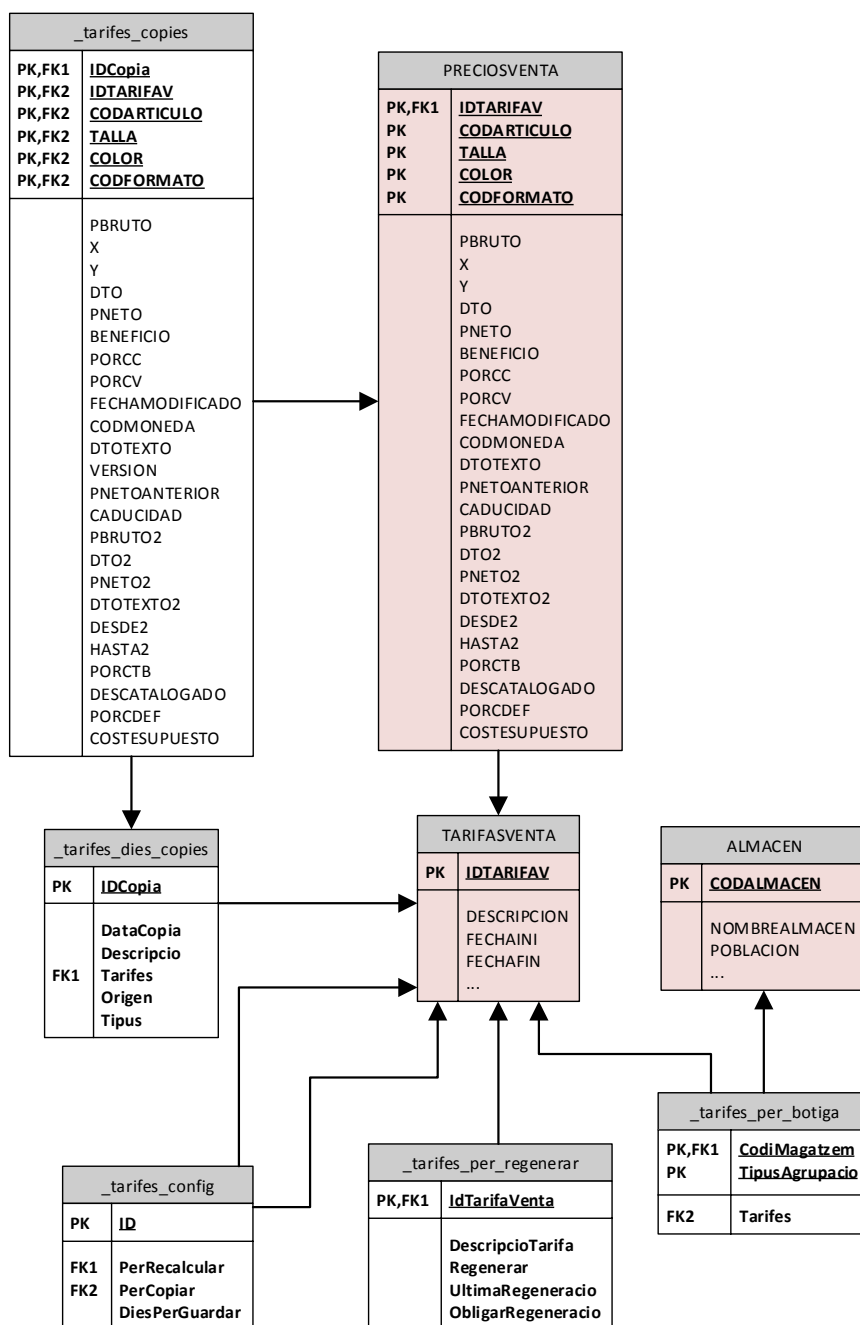




### 7.5.4 Taules per l'ús d'aplicacions relacionades amb les tarifes i preus de venda

Per l'objectiu del procés d'actualització dels preus dels diferents articles, cal tenir enregistrades quines tarifes cal regenerar (\_tarifes\_per\_regenerar), així com les que cal efectuar una còpia automàtica o recalculer-les (\_tarifes\_config). Per les diferents còpies que es generin cal registrar-ne la data i les tarifes que es copien (\_tarifes\_dies\_copies), així com els preus en si (\_tarifes\_copies) que és una rèplica exacta de la taula PRECIOSVENTA de l'ERP.

També cal emmagatzemar les tarifes associades a cada botiga segons diferents tipus d'agrupació que utilitzarem en certs aplicatius del sistema.



## 7.6 Implementació dels components del projecte

Per tal de fer factible la lectura i la visualització del desenvolupament dels diferents components, ens limitarem a anotar el mínim codi possible que ens serveixi d'exemple, i centrar-nos en la funcionalitat de cada component.

### 7.6.1 SetelSysCore

Al nucli del sistema, SetelSysCore, li correspondrà entre d'altres:

- Donar representació a les entitats de la capa de negoci de l'empresa, relacionada amb les bases de dades de l'ERP i la web comercial.
- Donar una representació i funcionalitat als objectes de la gestió del sistema.
- Dotar al sistema de les eines de gestió d'errors i servei de missatgeria.
- Registrar l'activitat i les connexions establertes autenticant-les correctament.
- ...

Anem a analitzar les diferents classes que el conformen, tot i que en el corresponent apartat d'[especificació i disseny](#) hem vist ja l'estructura de classes que el componen i les seves dependències.

#### *Classe Utils:*

En aquesta classe hem implementat tot un seguit de mètodes i funcions d'ús compartit i estàtic que podem utilitzar per funcions de cadena, operacions d'entrada / sortida amb fitxers, etc.

Alguns exemples:

```
Public Shared Sub VolcarResultatsToFitxer(Values As List(Of String), ArxiuSortida As String)
Public Shared Function NormalitzarCadena(Cad As String) As String
Public Shared Function ClonarLlista(Llista As List(Of String)) As List(Of
Public Shared Function FormatejarTamanyArxiu(Bytes As Long, Tipus As TipusTamany) As String
```

#### *Classe ExcelUtils:*

Implementa tot un seguit de mètodes i funcions d'accés estàtic que ens permet gestionar operacions de lectura i generació d'arxius d'Excel i separats per comes (CSV). Utilitza una llibreria externa (DTG.Excel) que anotem al [final](#) del capítol de desenvolupament. Posem un exemple de mètodes i funcions implementades:

```
Public Shared Function LlegirExcel(NomFitxer As String, NomFulla As String, TipusFitxer As TipusSortidaExcel, Optional TeHeaders As Boolean = True) As DataTable
Public Shared Function LlegirCSV(NomFitxer As String, Optional TeHeaders As Boolean = False) As DataTable
```

**Public Shared Function** LlegirCSV(NomFitxer *As String*, Separador *As String*, PrimeraFilaEsDescriptiva *As Boolean*, Codificacio *As System.Text.Encoding*) *As DataTable*

**Public Shared Function** GuardarGridToCSV(Graella *As Windows.Forms.DataGridView*, NomFitxer *As String*, GuardarNomsColumnes *As Boolean*, Codificacio *As System.Text.Encoding*) *As Boolean*

**Public Shared Sub** GuardarDTToExcel(Taula *As DataTable*, NomArxiuSortida *As String*, NomFullaSortida *As String*, TipusFitxer *As TipusSortidaExcel*)

**Public Shared Sub** GuardarGridToExcel(Graella *As System.Windows.Forms.DataGridView*, NomFitxer *As String*, NomFulla *As String*, VolcarOcultaes *As Boolean*, TipusFitxer *As TipusSortidaExcel*)

**Public Shared Function** DeterminarTipusArxiuExcelSegonsExtensio(NomArxiu *As String*) *As TipusSortidaExcel*

### *Classe ArxiuConfiguracio:*

Utilitzada per l'ús amb fitxers de configuració utilitzats per diferents components del programa. Citem a continuació la constructora i la funció per a la consulta d'un paràmetre existent en el fitxer:

**Public Sub New**(Ruta *As String*)

**Public Function** ConsultarParametre(Clau *As String*) *As String*

El fitxer ha de complir la següent estructura:

```
....
[Capçalera opcional]
Parametre1=Valor1
Parametre2=Valor2
...
```

### *Classe SQLServer:*

Aquesta classe s'encarrega de:

- Implementar la connectivitat amb les bases de dades físiques
- Implementar la lectura de definicions de bases de dades.
- Creació i eliminació de taules de la base de dades.
- Gestió de transaccions en bloc.

A l'instanciar un nou objecte d'aquesta classe, cal dotar al constructor de les dades de connexió necessàries per a connectar-se a la base de dades desitjada.

**Public Sub New**(Host *As String*, DB *As String*, User *As String*, Pwd *As String*, TestConnection *As Boolean*)

Al consultar les dades, ja sigui a través d'una comanda SQL en format cadena de text, o bé utilitzant la classe *SqlCommand*<sup>21</sup>, ens retornarà sempre un objecte del tipus *DataTable* amb el contingut fruit de la consulta que posteriorment adaptarem a través d'una rutina encarregada de convertir aquests resultats en entitats del sistema.

---

<sup>21</sup> SqlCommand Class – [Microsoft MSDN](#)

**Public Function** Consultar(Comm [As SqlCommand](#), TancarConexio [As Boolean](#)) [As DataTable](#)

Per modificar dades de la base de dades existents utilitzem alguna de les funcions següents, ambdós sobrecarregades en la definició:

**Public Function** Modificar(strSQL [As String](#), TancarConexio [As Boolean](#)) [As Long](#)

**Public Function** Modificar(Comm [As SqlCommand](#), TancarConexio [As Boolean](#)) [As Long](#)

**Public Function** ModificarEnBloc(strSQLs [As List\(Of String\)](#), TancarConexio [As Boolean](#)) [As Long](#)

**Public Function** ModificarEnBloc(Comms [As List\(Of SqlCommand\)](#), TancarConexio [As Boolean](#)) [As Long](#)

Finalment, per a la gestió de transaccions disposem d'un mètode per l'inici de la transacció i un per la finalització poden especificar si volem executar el "commit" sobre la base de dades. Pel que fa a les modificacions dins de les transaccions també hem definit un mètode sobrecarregat:

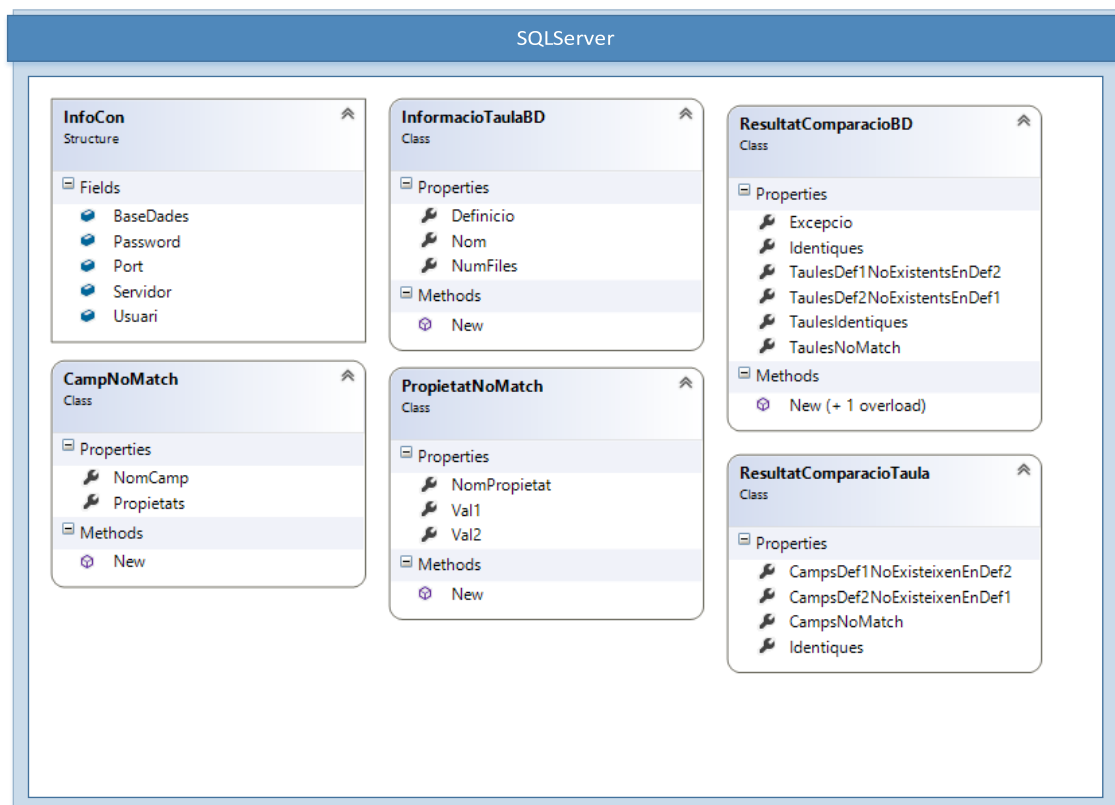
**Public Sub** IniciarTransaccio()

**Public Function** ModificarDinsTransaccio(Comm [As SqlCommand](#)) [As Integer](#)

**Public Function** ModificarDinsTransaccio(Comms [As List\(Of SqlCommand\)](#)) [As Integer](#)

**Public Sub** FinalitzarTransaccio(Commit [As Boolean](#))

Estructura de la classe:



### *Classe ComparadorCadenes:*

Implementa un algorisme de càlcul de distàncies entre dos textos, utilitzant l'algorisme de distàncies de Levenshtein<sup>22</sup>, a part de suplir els mètodes necessaris per l'estandardització de cadenes, que utilitzem per la comparació de valors introduïts ens els camps referents a la direcció d'un client.

Public Shared Function Comparar(Cadena1 As String, Cadena2 As String) As Integer

Public Shared Function Estandaritzar(Cadena As String) As String

Public Shared Function EstandaritzarCP(CP As String) As String

Public Shared Function DistanciaLevenshtein(ByVal Cadena1 As String, ByVal Cadena2 As String) As Integer

### *Classe Mercury:*

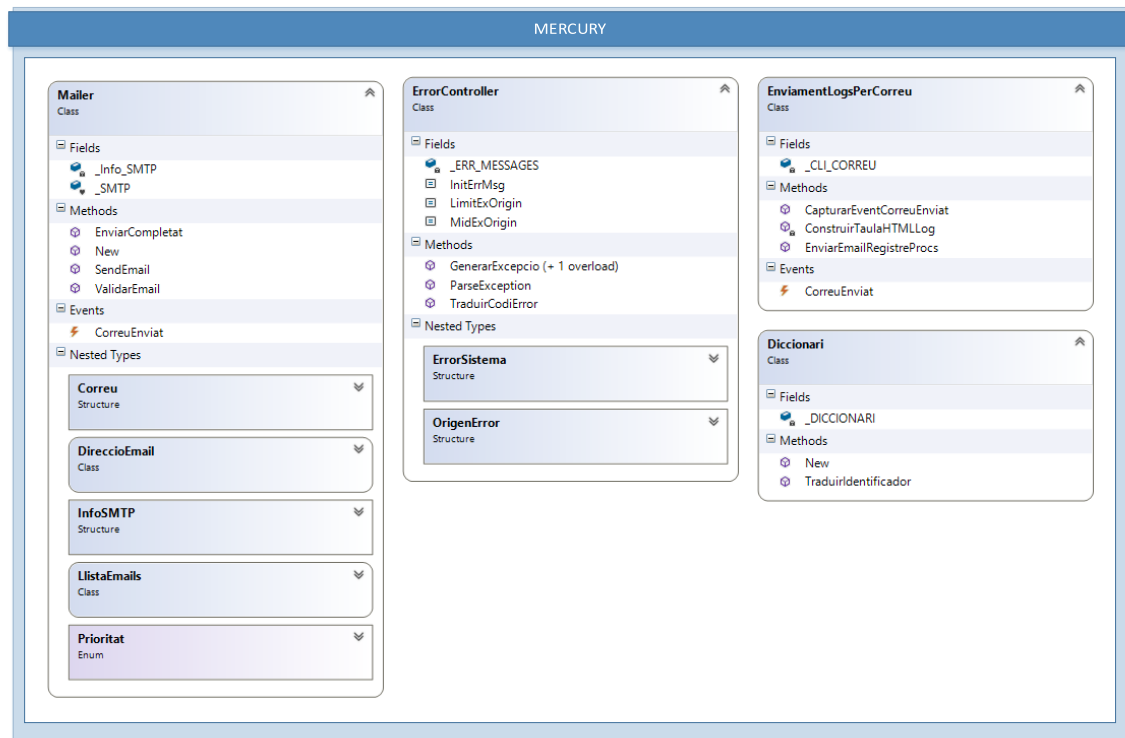
Implementa les funcionalitats següents:

- Encapsulament d'errors propis sota l'estructura **ErrorSistema**, i els mètodes necessaris per generar-los i interpretar-los. Tot això dins la classe **ErrorController**. Hem trobat adequat poder controlar en el nostre sistema d'errors, la traçabilitat del mateix, o sigui en quin objecte es genera i en quin mètode o funció fins que arriba a la classe encarregada de registrar l'error **Midgard**. Per fer-ho utilitzem l'estructura **OrigenError**. Més endavant veurem com es controla la traçabilitat dels errors.
- Generar els enviaments de correus de notificacions i informatius necessaris en diferents punts del sistema. Per fer-ho s'implementa la classe **Mailer**, encarregada de gestionar l'enviament d'aquests correus, així com el resultat de l'enviament fent ús de les classes Correu, **DireccioEmail**, **InfoSMTP**, **LlistaEmails** i l'enumeració **Prioritat**.
- Actuar com a sistema de traducció lingüística de recursos a través de la classe **Diccionari**, on tenim emmagatzemat un gestor de recursos al qual podem aplicar sistemes de traducció. Tot i això actualment el sistema és monolingüe però hem deixat el sistema obert per futures modificacions en l'escenari.

En el següent esquema podem veure el contingut de la classe Mercury, on apareixen les diferents classes citades anteriorment.

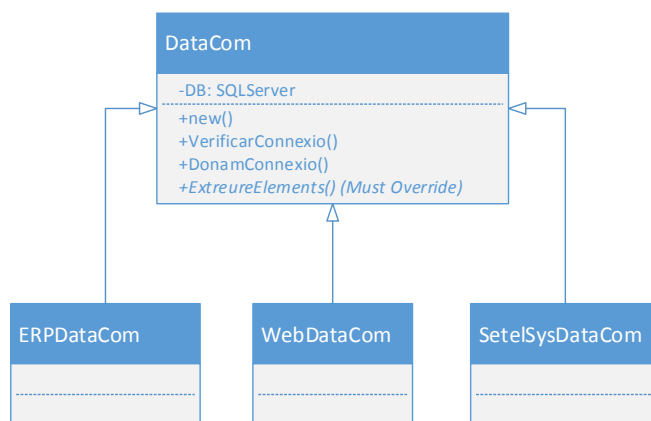
---

<sup>22</sup> Algoritmo de Levenshtein – [Algoritmos de similitud y distancia](#)



### Classe DataCom:

Classe de la qual posteriorment heretaran<sup>23</sup> els diferents comunicadors (ERPDataCom, SetelSysDataCom i WebDataCom), i s'encarrega de definir els elements comuns així com la constructora comuna associada a tot ells.



Constructora comuna en tots els comunicadors:

`Public Sub New(DadesConnexio As SQLServer.InfoCon)`

<sup>23</sup> Inheritance in Visual Basic .NET – [Microsoft MSDN](https://docs.microsoft.com/en-us/dotnet/visual-basic/inheritance)

Relació amb el connector de dades SQLServer, per verificar la connectivitat del comunicador i donar accés directe a la base de dades per consultes especialitzades que no estan compreses en els comunicadors. Més endavant veurem quines comprenen.

[Public Function](#) VerificarConnexio() [As Boolean](#)

[Public Function](#) DonamConnexio() [As SQLServer](#)

Definició de la funció que tots els comunicadors hauran d'implementar, destinada a l'extracció de dades corresponents a una entitat de la base de dades identificada dins l'enumeració **DB\_OBJECTS**:

[Public MustOverride Shared Function](#) ExtreureElements(TipusElement [As DataCom.DB\\_OBJECTS](#), Dades [As DataTable](#)) [As List\(Of Object\)](#)

L'enumeració de tipus DB\_OBJECTS conté una referència a les diferents taules físiques implementades en cada base de dades a la que associem els comunicadors heretats.

```
<Serializable(>
Public Enum DB_OBJECTS As Integer
    DBO_web_operaciones_tpv = 1000
    DBO_web_textos_it = 1001
    ...
    DBO_erp_ARTICULOSLIN = 2000
    DBO_erp_ARTICULOS = 2001
    ...
    DBO_setelsys_processos = 3000
    DBO_setelsys_usuaris = 3001
    ...
End Enum
```

*Classe ERPDataCom:*

Com hem vist en el punt anterior aquesta és una de les classes que hereta de **DataCom**, i que en aquest cas s'encarregarà de donar comunicació amb les entitats del sistema ERP (articles, tarifes, preus, ventes, compres,...).

En tots els comunicadors hem utilitzat la mateixa estratègia, consistent en definir una representació de cada un dels elements existents en el sistema a través d'una classe que l'identifica i ens dona els mètodes més comuns per la consulta, modificació i inserció.

```
<Serializable(>
Public Class DB_Articulo
    Public Property CODARTICULO As Integer
    Public Property DESCRIPCION As String
    Public Property DESCRIPADIC As String
    Public Property REFPROVEEDOR As String
    ...

    Public Shared Function Clonar(Orig As DB_Articulo) As DB_Articulo
    Public Shared Function GenerarComandaActualitzarArticle(Article As DB_Articulo) As SqlCommand
    Public Shared Function GenerarComandaInsercioArticle(Article As DB_Articulo) As SqlCommand
    Public Shared Function GenerarComandaEsborratArticle(Codi As Integer) As SqlCommand
    Public Shared Function ConsultarArticle(Codi As Integer) As DB_Articulo
    ...
End Class
```

D'aquesta manera dins de cada classe representativa d'una entitat de la base de dades:

- **Encapsulem** la seva informació a nivell d'atributs en una classe amb el mateix nom i atributs que en la base de dades, per així ser més fàcil i estàndard accedir a les dades.
- Implementem el màxim de comandes del tipus **CRUD** <sup>24</sup> dins de la classe representativa per facilitar la **reutilització** del codi i l'**accessibilitat**.
- En les comandes que impliquen modificacions de la base de dades retornem sempre objectes del tipus **SqlCommand**, mai modifiquem directament la base de dades, perquè posteriorment en els gestors d'aplicacions sovint es volen executar comandes dins de transaccions. Retornant aquest objecte, **desvinculem** els gestors d'aplicacions amb l'accés a codi SQL i li passem directament la comanda que caldrà executar sobre la base de dades, amb o sense **transacció** associada.

Finalment també cal implementar la funció declarada en la classe **DataCom** de la qual hereta, per poder extreure els elements. L'objectiu d'aquesta funció és agrupar el codi destinat a l'extracció de dades dels objectes **DataTable** que ens retorna la funció *Consultar()* de la classe **SQLServer**.

<sup>24</sup> Create, read, update and delete - [Wikipedia](https://en.wikipedia.org/wiki/CRUD)



L'estructura d'aquesta funció és la següent:

```
Public Shared Function ExtreureElements(TipusElement As DB_OBJECTS, Dades As DataTable) As List(Of Object)
    Try
        Select Case TipusElement

            Case DB_OBJECTS.DBO_erp ARTICULOS
                Dim res As New List(Of Object)

                For Each r As DataRow In Dades.Rows
                    Dim lin As New DB_Articulo
                    If r.IsNull("CODARTICULO") Then lin.CODARTICULO = Nothing Else lin.CODARTICULO = r.Item("CODARTICULO")
                    If r.IsNull("DESCRIPCION") Then lin.DESCRIPCION = Nothing Else lin.DESCRIPCION = r.Item("DESCRIPCION")
                    If r.IsNull("DESCRIPADIC") Then lin.DESCRIPADIC = Nothing Else lin.DESCRIPADIC = r.Item("DESCRIPADIC")
                    ...
                    res.Add(lin)
                Next

                Return res

            Case DB_OBJECTS.DBO_erp ARTICULOSLIN
                ...
                El mateix procediment però per l'objecte "DB_Articuloslin"
                ..

            Case Else
                Throw Mercury.ErrorController.GenerarExcepcio(ERR_EXTRACCIO_ELEMENTS)
                Exit Select

        End Select

    Catch ex As Exception
        Throw Mercury.ErrorController.GenerarExcepcio("SetelSysCore", "WebDataCom", "ExtreureElements", ex)
    End Try
End Function
```

Utilitzem les característiques del **polimorfisme**<sup>25</sup> per encapsular el resultat en objectes generals que després podrem reinterpretar en la corresponent comanda que ha fet la crida de la funció per usar-los com l'objecte que pertoca.

L'ús d'aquesta funció ens permet aïllar totes les comandes d'extracció de camps en un sol punt per maximitzar la **facilitat** en la creació de nou codi i poder **reutilitzar** sempre aquesta lectura de camps directament dependent de les taules de la base de dades.

### *Classes SetelSysDataCom i WebDataCom:*

Tot el que hem explicat sobre la classe **ERPDataCom**, s'aplica sobre aquestes dos classes, amb la diferència que cada una s'especialitza en focalitzar-se en entitats de bases de dades diferents.

- SetelSysDataCom → Base de dades de SetelSys (Usuaris, botigues, aplicacions, ...)
- WebDataCom → Base de dades de la web comercial (textos, preus, promos, packs,...)

<sup>25</sup> How Visual Basic Provides Polymorphism – [Microsoft MSDN](#)

### *Classe Midgard:*

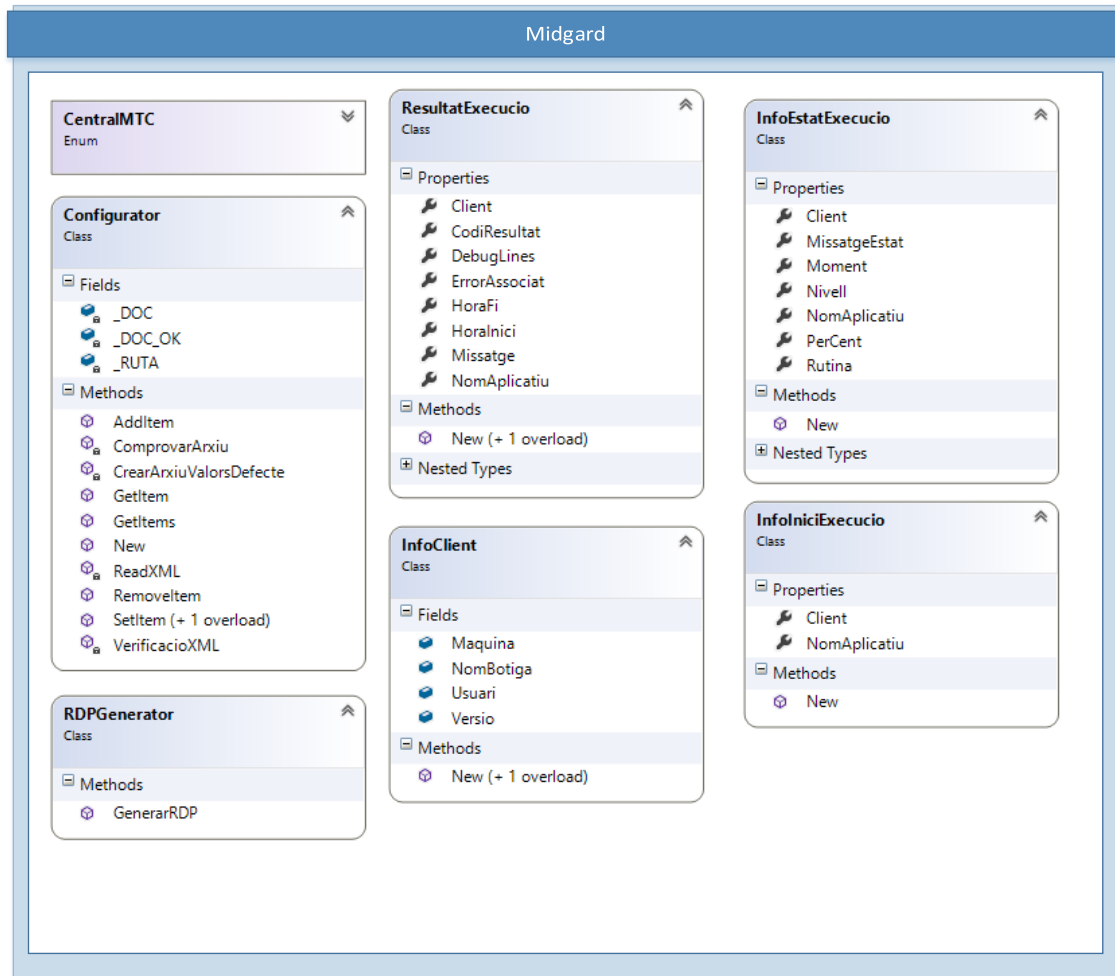
La classe Midgard és l'encarregada de gestionar totes les connexions entrants amb el sistema. Per dir-ho d'alguna manera és la porta d'entrada al nucli. Les funcions que té les podem resumir en:

- Carregar les dades de connexió a les diferents bases de dades definides en l'arxiu "config.xml" existent en la carpeta de la solució. La lectura d'aquest fitxer es fa a través de la classe **Configurator**<sup>26</sup> especialitzada per aquest objectiu.
- Generar els arxius de RDP associats a les botigues i usuaris, a través de la classe **RDPGenerator**.
- Captar la informació de la connexió (Màquina, nom de la botiga, usuari i versió en ús) a través de la classe **InfoClient**.
- Gestionar l'execució de processos automatitzats capturant-ne tots els events encapsulats sota les classes **InfoIniciExecucio**, **InfoEstatExecucio** i **ResultatExecucio**. S'encarrega també de guardar l'estat de dites execucions sobre el registre d'activitat de la base de dades al rebre els corresponents events definits en el Midgard
- Distribuir les connexions als diferents **comunicadors** ERPDataCom, SetelSysDataCom i WebDataCom.
- Registrat tots els errors transmesos des de les diferents classes d'origen a la base de dades en el registre corresponent d'errors del sistema, encapsulats sota la classe **ErrorSistema**.
- Atorgar accés als diferents comunicadors perquè en puguin fer ús els gestors d'aplicacions, desvinculant els gestors pròpiament de la resta de tasques més genèriques del sistema anotades en els punts anteriors.
- Implementar les rutines i funcions definides en els diferents comunicadors, estructurades en diferents classes definides temàticament, per poder fer més accessible l'accés a les dades sense implicar un accés directe als comunicadors, tot i que aquest accés seguirà sent viable per els gestors d'aplicacions que requereixin usos més avançats.

Veiem en el següent diagrama l'estructura de classes internes al Midgard i que hem referenciat fa un moment:

---

<sup>26</sup> Parse XML using XmlReader Class – [Microsoft MSDN](#)



De la mateixa manera es creen interfícies d'accés a les funcions més utilitzades dels comunicadors, per reutilitzar el màxim el codi i fer-lo més fàcilment accessible. Vegem-ne un exemple:

```
Public Class OpsERP
```

```
Public Logística As New OpsLogística
```

```
Public Articles As New OpsArticles
```

```
Public Tarifes As New OpsTarifes
```

```
Public Clients As New OpsClients
```

```
Public Proveïdors As New OpsProveïdors
```

```
...
```

```
Public Class OpsProveïdors
```

```
Public Function ConsultarProveïdors(OnConsultar As CentralMTC, ...) As List(Of DB_Proveïdor)
```

```
Public Function ConsultarProveïdor(Codi As Integer, OnConsultar As CentralMTC) As DB_Proveïdor
```

```
...
```

```
End Class
```

```
Public Class OpsClients
```

```
Public Sub ActualitzarDireccioClient(InfoClient As DB_Cliente,...)
```

```
Public Sub ActualitzarDireccioFitxaClient(CodiClient As Integer, Poblacio As String, ...)
```

```
Public Sub ActualitzarDireccioEnvioClient(CodiClient As Integer, CodiEnvio As Integer,...)
```

```
...
```

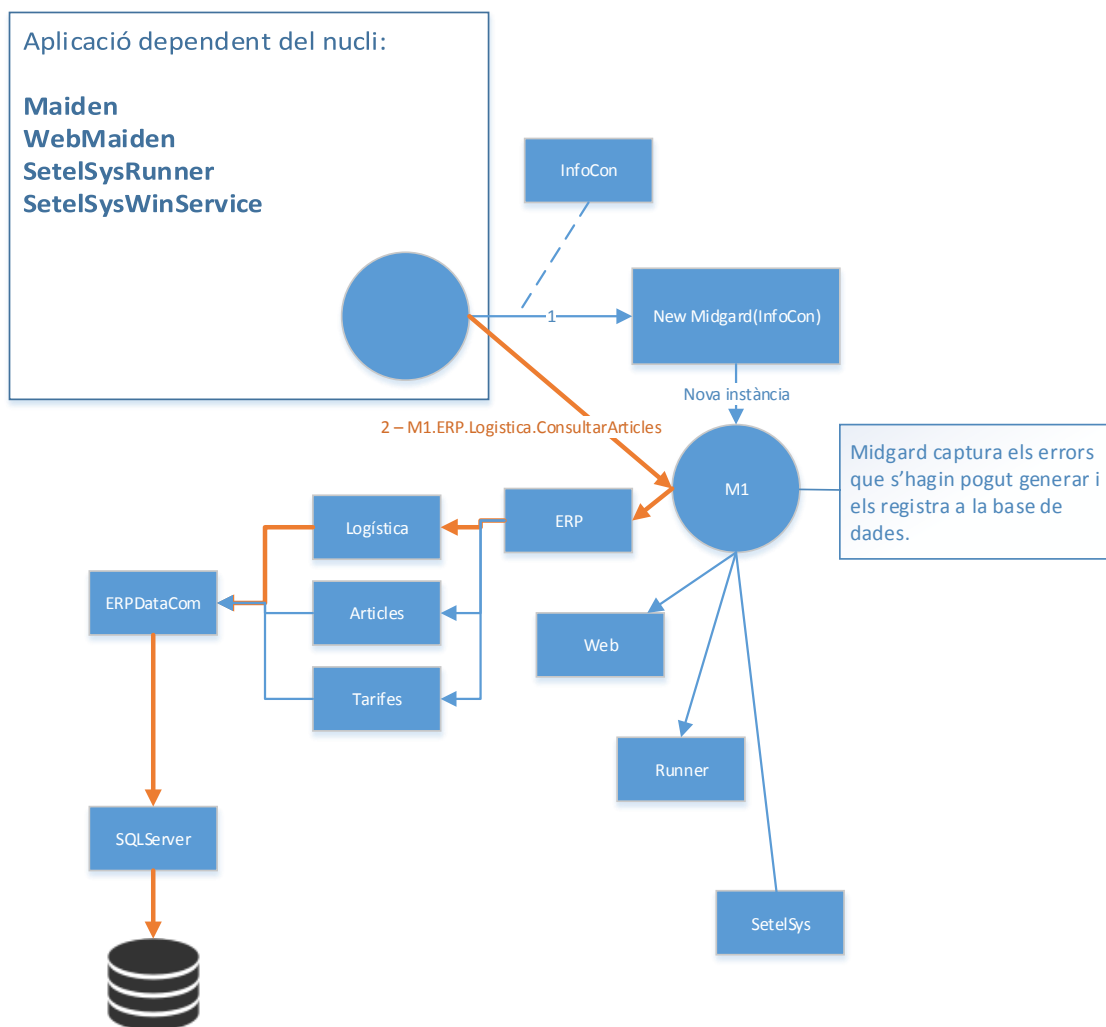
```
End Class
```

```
End Class
```

I al crear una nova instància del Midgard s'inicialitzen aquestes classes per donar accés a través de la instància als mètodes globals i genèrics dels comunicadors:

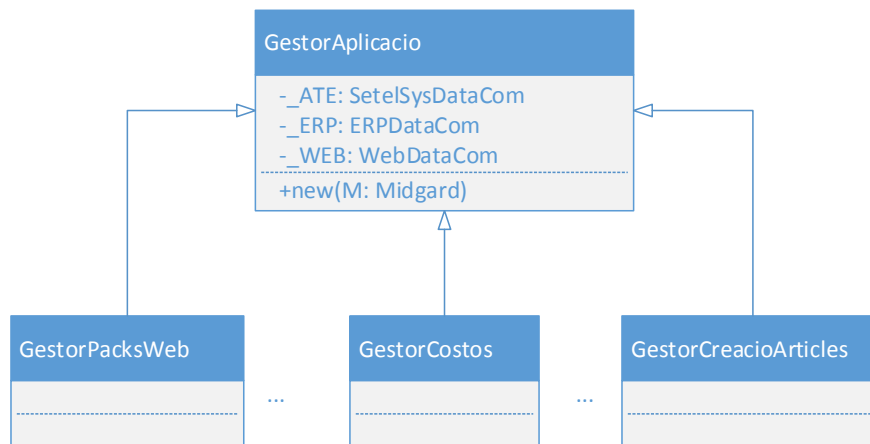
Public Web As New OpsWeb  
Public ERP As New OpsERP  
Public SetelSys As New OpsSetelSys  
Public Runner As New OpsRunner  
Public Logger As New OpsLogs  
Public Sistema As New OpsSistema

Tot seguit mostrem un diagrama on es pot visualitzar més clarament el procés que es genera a l'establir una connexió amb el nucli:



### *Gestors d'aplicacions especialitzades*

En alguns casos, no són suficients les operacions CRUD ofertes pels comunicadors, i cal desenvolupar un gestor d'aplicació que a través del Midgard i la potència bàsica dels seus comunicadors, pugui implementar funcionalitats més extenses. Per això hem creat la classe **GestorAplicacio**, de la qual heretaran els diferents gestors que facin falta afegir per incrementar funcionalitat del sistema:



Anotem un exemple de codi que demostra com podem utilitzar els gestors d'aplicacions vinculant-los amb el **Midgard**:

```

'Instanciem una connexió amb el Midgard:
Dim m As New Midgard(Client, CentralBotiga(Client.NomBotiga), True)

'Instanciem un gestor de packs per la web, assignant-li el Midgard instanciat
prèviament.
Dim g As New GestorPacksWeb(m)

'Executem la funció corresponent del gestor, que internament utilitzarà els
comunicadors per executar les comandes bàsiques sobre les entitats.
Return g.CrearPromoDunPack(IdPack, NomPromo, IdPromo)
  
```

### *Classe SesHat:*

Aquesta classe és l'encarregada de gestionar la fabricació **d'informes** per a poder ser descarregats des de la web en qualsevol dels formats disponibles (PDF, CSV o XLS).

S'han generat informes a través de dos motors diferents en base de les necessitats visuals i funcionals dels llistats. Un dels motors utilitzats és el propi motor integrat en l'entorn de desenvolupament de Microsoft Visual Studio (Microsoft Reports<sup>27</sup>) i l'altre motor utilitzat és el de Crystal Reports (Crystal Reports for Visual Studio<sup>28</sup>).

<sup>27</sup> Microsoft Reports – [Microsoft MSDN](#)

<sup>28</sup> Crystal Reports developer version for Visual Studio – [SAP.COM](#)

Exemple d'informe elaborat amb els motor d'informes de Microsoft:

Listado de albaranes generados: [ @TipusLlistat ]					[ @Data ]	[ @Hora ]
Alm.	Albaran	Cod. Cli.	Nombre		Fecha	Hora
[Magatz]	[Document]	[CodiClient]	[NomClient]		[Data]	[Hora]

Exemple d'informe elaborat utilitzant Crystal Reports:

Section1 (Encabezado del informe )																		
Listado de Artículos regularizados entre: ?Dates																		
Tipos regularización: ?TipusDocs																		
Artículo: ?Article																		
Marca: ?Marca																		
Familia: ?Familia																		
Section2 (Encabezado de página )																		
GroupHeaderSection1 (Encabezado de grupo #1: Linies.Magatzem - A )																		
Magatzem																		
GroupHeaderSection2 (Encabezado de grupo #2: Linies.Familia - A )																		
Familia																		
GroupHeaderSection3 (Encabezado de grupo #3: Linies.Marca - A )																		
Marca																		
Referencia	Descripción							Color	Talla	Ud.	Coste	Total	Fecha					
Section3 (Detalles )																		
Referencia	Descripioio							Color	Talla	Unitats	Cost	Total	Data					
GroupFooterSection3 (Pie de grupo #3: Linies.Marca - A )																		
Total							Nombre de gru		Linies	s.Total								
GroupFooterSection2 (Pie de grupo #2: Linies.Familia - A )																		
Total							Nombre de gru		Linies	s.Total								
GroupFooterSection1 (Pie de grupo #1: Linies.Magatzem - A )																		
Total							Nombre de gr		Linies	s.Total								
Section4 (Pie del informe )																		
Total general:							Linies		s.Total									
Section5 (Pie de página )																		
Página N de M																		

Els informes s'han vinculat directament amb classes definides dins del projecte especialment per adaptar les dades que es volen mostrar, per exemple aquí veiem com s'associen alguns informes directament a classes creades per adaptar les dades a l'informe:

Informe	Classe lligada
repAlarmesMaterial.rpt	repLiniaAlarmes.vb
repAlbaransGeneratsSP.rdlc	repLiniaAlbaraGeneratSP.vb
repCaixesMatBotigues.rpt	relliniaCaixesMatBotigues.vb
repCanvisFicticisReals.rpt	repLiniaCanviFicticiReal.vb
...	...

Per a generar un informe, s'utilitza una funció estàtica dins del **Seshat**:

`Public Shared Function GenerarReport(Tipus As TipusReport, Format As FormatSortida, Linies As IEnumerable(Of Object), Optional Params As Array = Nothing) As String`

En la funció s'especifica el tipus d'informe, associat a una enumeració **TipusReport** i el format de sortida del document associat a una altra enumeració **FormatSortida**. S'especifiquen les línies de dades que han d'emplenar l'informe, que com es pot observar es declaren com una **llista** enumerable **d'objectes** (perquè es pugui utilitzar la mateixa funció per a tots els informes). Finalment es poden assignar també paràmetres addicionals en un vector que poden ser requerits per alguns informes. La funció retorna la **ruta** a l'arxiu generat.

Formats possibles especificats per l'enumeració **FormatSortida**:

```
Public Enum FormatSortida As Integer
```

```
PDF = 1
```

```
Excel = 2
```

```
CSV = 3
```

```
End Enum
```

Possibles tipus d'informes especificats a l'enumeració **TipusReport**:

```
Public Enum TipusReport As Integer
```

```
ArticlesCreatesManager = 1
```

```
MaterialPerRebre = 2
```

```
PedidosPendentsAmpliat = 3
```

```
MaterialPerDemandar = 4
```

```
....
```

```
End Enum
```

Per veure com associa aquesta funció als informes dissenyats, podem veure aquest tros de codi clarificador:

```
Public Shared Function GenerarReport(Tipus As TipusReport, Format As FormatSortida,  
    Linies As IEnumerable(Of Object), Optional Params As Array =  
    Nothing) As String
```

```
Try
```

```
Dim Ruta As String = "c:\inetpub\wwwroot\SetelSys\tmp\"
```

```
Dim NomFitxer As String
```

```
Select Case Tipus
```

```
Case TipusReport.CaixasMatBotigues
```

```
Dim Rep As New repCaixesMatBotigues
```

```
Rep.SetDataSource(Linies)
```

```
NomFitxer = "cmb" & TimeStamp() & GetExtensioFormatExportacio(Format)
```

```
Rep.ExportToDisk(GetFormatExportacio(Format), Ruta & NomFitxer)
```

```
Return NomFitxer
```

```
Case ....
```

```
Resta de casos...
```

```
Case Else
```

```
Return ""
```

```
End Select
```

```
Catch ex As Exception
```

```
Throw Mercury.ErrorController.GenerarExcepcio("SetelSysCore", "Seshat",  
"GenerarReport", ex)
```

```
End Try
```

```
End Function
```

Instanciem la classe de l'informe

Assignem les línies de dades

Generem el nom de fitxer

Exportem el fitxer en el format corresponent

Per finalitzar mostrem un exemple de codi per entendre el procés d'adaptació de les dades provinents d'objectes no afins als de les línies de l'informe, en un exemple de generació d'un informe:

```
Public Function GenerarInformeCaixesMaterialCentral(
    Format As Seshat.FormatSortida,
    Caixes As List(Of SetelSysDataCom.Caixa),
    ConsultarExtraccions As Boolean,
    DataInici As DateTime,
    DataFi As DateTime)

    Try
        Dim Línies As List(Of repLiniaCaixesMatCentral)
        Línies = GenerarLíniesInformeCaixesCentral(Caixes, ConsultarExtraccions, DataInici, DataFi)
        Return Seshat.GenerarReport(Seshat.TipusReport.CaixesMatCentral, Format, Línies)
    Catch ex As Exception
        Throw GenerarExcepcio("Heimdall", "ERPDataCom", "GenerarInformeCaixesMaterialCentral", ex)
    End Try

End Function
```

En la funció *GenerarLíniesInformeCaixesCentral*, és on es transforma la informació d'entrada, en la corresponent informació necessària per a generar l'informe que ens fa falta.

### Gestió dels processos automatitzats

L'encarregat de gestionar els processos automatitzats és la classe **Midgard**. En concret la funció *ExecutarProces*, fent ús de les classes vistes prèviament: **InfoIniciExecucio**, **InfoEstatExecucio** i **ResultatExecucio**:

```
Public Function ExecutarProces(Peticio As InfoIniciExecucio) As ResultatExecucio
    Dim Inici As DateTime = Now, Dim Fi As DateTime = Now
    Dim res As ResultatExecucio, Dim DebugLines As New List(Of String)

    Try
        Select Case Peticio.NomAplicatiu
            Case "BorrarEtiquetas"
                _ERP.BorrarEtiquetas(Peticio, DebugLines)
                Fi = Now
            Case ...
                Resta de casos de procesos...
            Case Else
                Throw GenerarExcepcio(PROCES_NO_MAPEKJAT)
        End Select
        res = New ResultatExecucio(Peticio, Inici, Fi, DebugLines)
        Return res

    Catch ex As Exception
        Fi = Now
        res = New ResultatExecucio(Peticio, Inici, Fi, DebugLines,
            GenerarExcepcio("Midgard", "SetelSysCore.OpsRunner", "ExecutarProces", ex))
        Return res
    End Try

End Function
```



Com es pot veure en tots els processos es registra l'hora i data d'inici i finalització i el resultat, ja sigui correcte o erroni.

Els processos tenen l'opció d'utilitzar diferents events definits en el **Midgard** per poder comunicar l'estat d'execució a les classes superiors i poder-se registrar correctament tots els esdeveniments al notificar-ho al Midgard, que a la seva vegada torna l'event a la capa d'interfície gràfica.

Podem veure tot seguit, alguns exemples de llançaments d'events<sup>29</sup> per part d'un procés:

1. Inici del procés, informant amb l'objecte *Peticio* (de la classe **InfoIniciExecucio**) del seu inici:

```
RaiseEvent IniciExecucio(Peticio)
```

2. Comunicació de l'estat del procés, anotant en nom, el nivell del procés (global o parcial), el percentatge, el missatge assignat, informació del client i data i hora:

```
RaiseEvent EstatExecucio(New InfoEstatExecucio(Peticio.NomAplicatiu,
InfoEstatExecucio.TipusNivell.ProcesGlobal, 0, "Reasignando stock de los
pedidos temporales...", Peticio.Client, Now))
```

3. Comunicació de la finalització del procés, informant amb la informació de la petició d'execució inicial, hora d'inici, i de finalització, així com si el resultat ha estat satisfactori (*True / False*) i l'excepció capturada en cas d'haver-se produït:

```
RaiseEvent FinalitzacioExecucioProces(Peticio, HoraIni, Now, True,
Nothing, DebugLines)
```

### *Traçabilitat dels errors*

Hem considerat molt important canalitzar al màxim els errors, registrar-los i guardar-ne la traçabilitat per poder trobar l'origen de dit error en cas de produir-se. En totes les funcions i mètodes definits al llarg del codi, s'han utilitzat les comandes **Try/Catch**<sup>30</sup> de forma que es capturessin sempre tots els errors i s'encapsulessin dins un objecte personalitzat amb el contingut de l'error i la traça acumulada: **ErrorSistema**, definit en la classe **ErrorController** del component **Mercury**.

```
Try
    ... codi ...
Catch ex As Exception
    Throw GenerarExcepcio("ERPDataCom", "DB_Clientes", "ConsultarClient()", ex)
End Try
```

<sup>29</sup> Handling and raising events on .NET – [Microsoft MSDN](#)

<sup>30</sup> Try / Catch Syntax on VB .NET – [Microsoft MSDN](#)

Com podem veure en el codi anterior, la funció *GenerarExcepcio* està implementada en la classe **ErrorController**:

```
Public Shared Function GenerarExcepcio(NomComponent As String, NomClasse As String, NomRutina As String, ExcepcioInterna As Exception) As Exception
```

També podem generar excepcions de forma controlada per notificar errors de coherència o situacions detectades no esperades però que podem notificar un error registrat. En aquest cas fem servir la comanda:

```
GenerarExcepcio(CodisError.ID_ERROR)
```

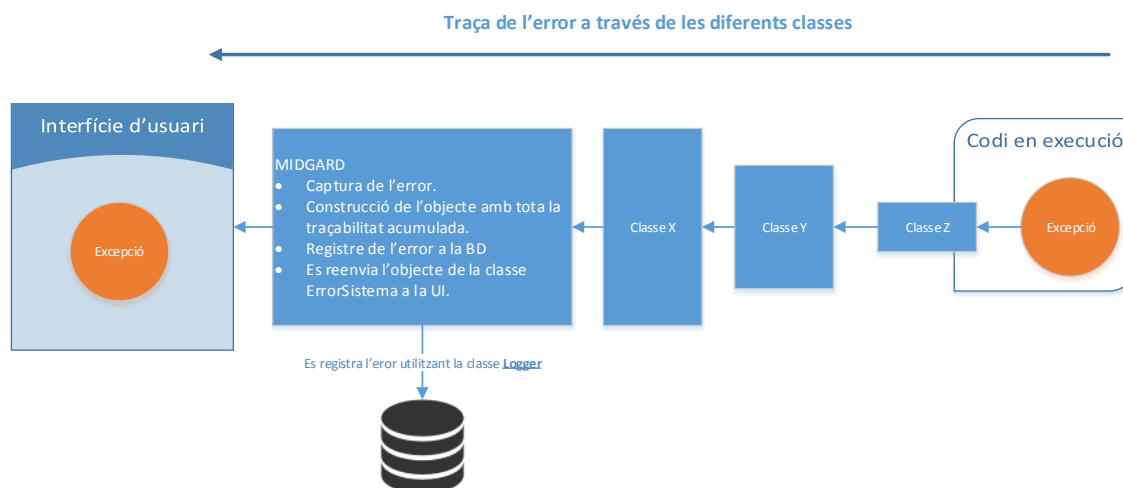
Que està definida implementada també en la classe **ErrorController**:

```
Public Shared Function GenerarExcepcio(Codi As CodisError) As Exception
```

On *CodisError* és una enumeració destinada a emmagatzemar els identificadors d'error que tenen també un missatge textual associat, per facilitar posteriorment les tasques associades a un futura implantació multilingüe.

```
Public Shared Function TraduirCodisError(Codi As CodisError) As String
```

En el següent esquema es pot veure el viatge de l'excepció fins al Midgard, així com les accions preses per aquest al rebre l'excepció:



### *Serialització de classes*

Per tal de poder enviar els objectes instanciats de diferents classes del sistema a través dels serveis web, fa falta definir-los com a **Serialitzables**<sup>31</sup> utilitzant la clàusula següent que hem aplicat en la definició de totes les classes sensibles a ser enviades pels serveis:

```
<Serializable()>
Public Class NomClasse
    ...
    ...
End Enum
```

### *Donant lloc al mite: Noms utilitzats pels components globals del sistema*

Com a nota curiosa us volíem explicar el motiu dels noms utilitzats en alguns dels components del nucli del SetelSysCore. Simplement, podeu fixar-vos que la majoria són noms de figures o indrets mitològics de diferents cultures. Us anotem una llista dels noms dels components i quines figures i llocs mitològics tenen associats/des:

Nom	Descripció
<b>Midgard</b>	En la mitologia escandinava Midgard és un dels mons de la realitat. Concretament aquell en el que viuen els humans. Està connectat a Asgard per Bifrost.
<b>Mercury</b>	Mercuri (en llatí Mercurius), déu de la mitologia romana, era el missatger dels déus, protector del comerç, fill de Júpiter i Maia Maestas.
<b>Seshat</b>	Deessa egípcia de la ciència, literatura, història, biblioteques, arquitectura i matemàtiques, i filla o muller de Thoth. En l'art era representada coma una humana amb un estel sobre el cap, vestint una bata de pell de lleopard.

<sup>31</sup> Serialization (C# and Visual Basic) – [Microsoft MSDN](#)

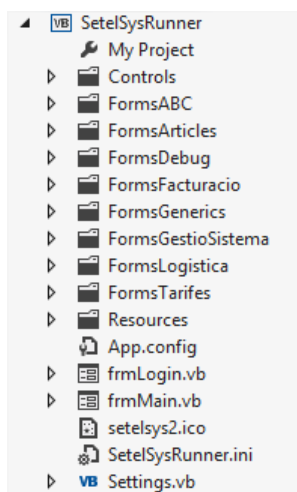
### 7.6.2 SetelSysRunner

SetelSysRunner ha estat desenvolupat a través de WinForms<sup>32</sup> (formularis per a Windows), com a eina de control administratiu del sistema i de parametrització de les diferents aplicacions existents en la interfície d'usuari SetelSys feta amb Silverlight. Recordem breument, tot i haver-se anotat anteriorment, els objectius d'aquesta aplicació administrativa:

- Gestionar els **processos** automatitzats existents (afegir, editar, eliminar,...).
- Visualitzar i operar amb el **registre** d'activitat dels processos automatitzats.
- Gestionar les **entitats** del sistema SetelSys: Usuaris, permisos, botigues, aplicacions, notificacions,...
- **Parametritzar** processos complexos. Estem parlant en tot moment de parametritzacions que només cal poder editar puntualment per canviar paràmetres globals del funcionament de les diferents aplicacions.
- Dotar **d'eines** de consulta per als **desenvolupadors** a nivell intern i privat (activables només amb una combinació de tecles).

### Estructura del projecte .NET

En la següent figura mostrem com s'estructura el projecte .NET del SetelSysRunner:

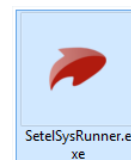


- **Controls:** Conté els diferents controls d'usuari desenvolupats per mostrar elements d'elaboració complexa i ús repetitiu en la interfície.
- Totes les carpetes que comencen amb "**Forms**" contenen formularis gràfics per la interacció amb l'usuari. Mes endavant citarem un inventari dels diferents formularis i quin objectiu tenen.
- **Resources:** Conté tots els arxius d'imatges utilitzades en l'aplicació per als botons, icones de notificació, quadres d'imatge,...

<sup>32</sup> Windows Forms – [Microsoft MSDN](https://docs.microsoft.com/en-us/dotnet/framework/winforms/)

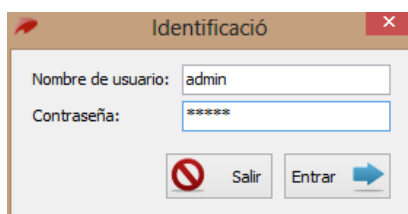
- **frmLogin.vb**: Implementa el procés de validació de l'usuari a l'iniciar el programa.
- **frmMain.vb**: Pantalla principal de treball de l'aplicació, des d'on podem accedir a tot el contingut de la resta de formularis.
- **SetelSysRunner.ini**: Arxiu de configuració necessari per l'arrancada del programa on s'especifica la seu de connexió.

La compilació d'aquest projecte genera un arxiu **executable**, dependent del nucli principal **SetelSysCore**, a través del qual executa tota la comunicació amb les dades del sistema SetelSys i del propi ERP.

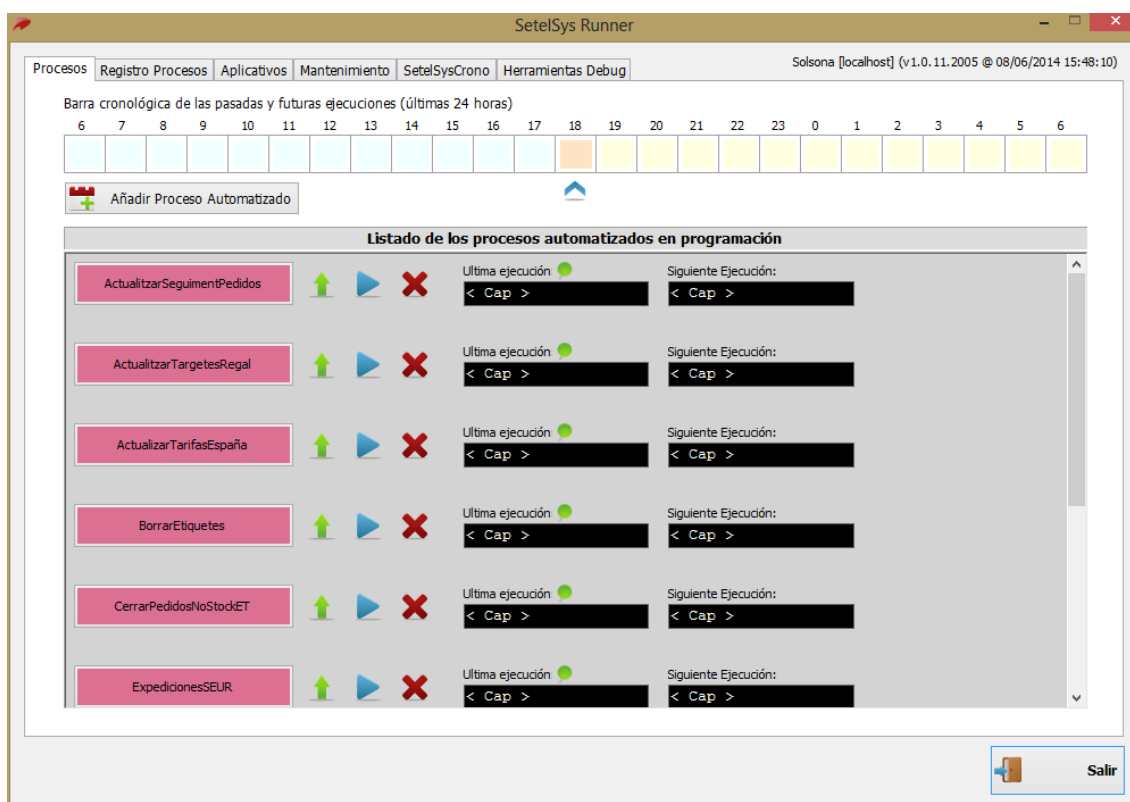


### Autenticació de l'usuari

A l'arrancar l'aplicació, el primer missatge que ens apareix és per validar l'usuari. Només l'usuari administrador poc accedir a l'aplicatiu.



Una vegada validat l'usuari administrador, ens apareix la pantalla principal del SetelSysRunner, des d'on podem accedir a varies categories d'opcions distribuïdes en pestanyes:



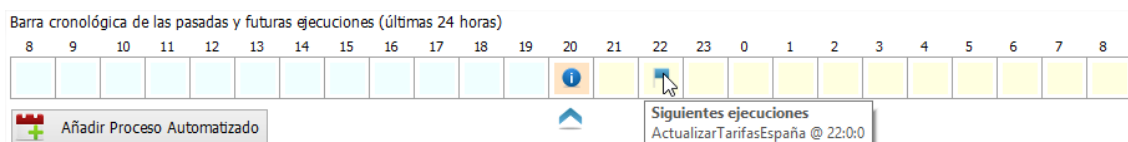
Anotem una breu descripció de la funcionalitat de cada pestanya:

- **“Procesos”**: Des d’on podem afegir, desactivar, editar, eliminar o executar manualment qualsevol procés automatitzat del sistema.
- **“Registro Procesos”**: Podem visualitzar el registre dels processos en un històric acumulat de totes les execucions passades, per poder veure si s’han executat correctament o amb error, i en aquest cas poder visualitzar en quin punt i rutina s’ha generat l’error i amb quin missatge.
- **“Aplicativos”**: Podem accedir a uns petits aplicatius per gestions sobre l’ERP. Són aplicatius molt especialitzats i que s’ha decidit de no publicar en l’aplicació web de Silverlight SetelSys, per evitar problemes innecessaris.
- **“Mantenimiento”**: Des d’on podem gestionar les parametritzacions dels processos, dels aplicatius de la web i del propi sistema.
- **“SetelSysCrono”**: Per configurar alguns paràmetres del servei, sobretot pel que fa a la gestió dels registres d’esdeveniments i el titular de notificació dels correus a enviar.
- **“Herramientas Debug”**: Pestanya especial amb les botoneres desactivades i només accessibles si s’utilitza una combinació de tecles especials “Ctrl+Alt+D”. Permet utilitzar un seguit d’eines de test.

### Gestió de processos automatitzats (Pestanya “Procesos”)

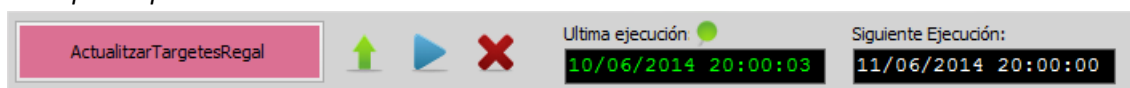
L’objectiu d’aquesta pestanya es poder visualitzar ràpidament la informació sobre execucions dels diferents processos, així com l’estat en el que estan.

A través d’una barra cronològica de les últimes 24 hores, podem visualitzar quines execucions s’han efectuat i si han donat algun tipus d’error o no:

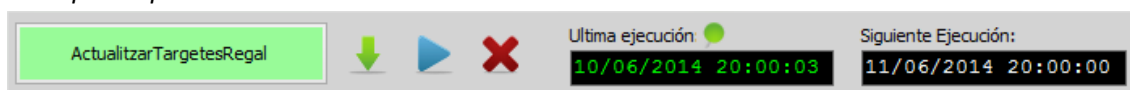


Podem visualitzar ràpidament si els processos estan actius perquè s’executin a l’hora programada (de color verd), o estan inactius (de color vermell). En cas d’estar inactius, encara que tinguin hora programada, no s’executaran.

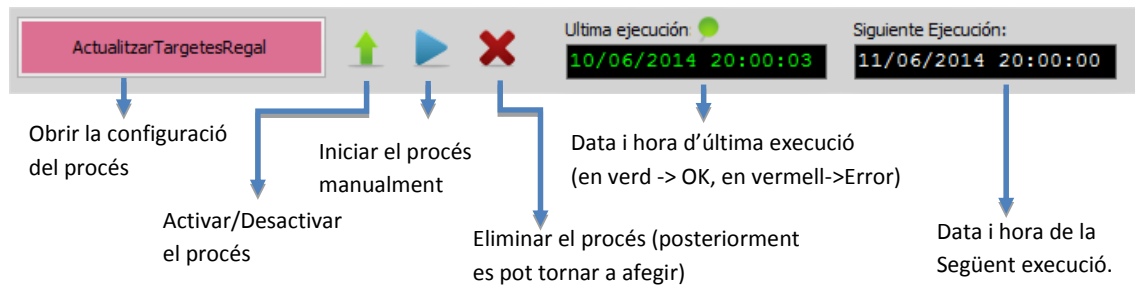
*Exemple de procés inactiu:*



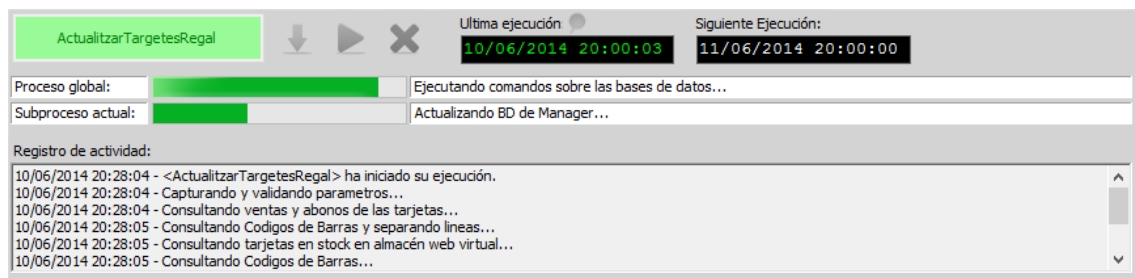
*Exemple de procés actiu:*



Les diferents accions que podem fer amb un procés és podem veure en el següent diagrama:



Podem forçar l'execució de qualsevol dels processos des d'aquest control, i quan s'inicia apareixen controls extres destinats a mostrar el procés global i parcial de l'execució, emplenant el corresponent quadre de text inferior amb el registre d'activitat del procés.



La informació referent al registre d'activitat, s'ha traspassat cap a la capa gràfica a través de la programació d'esdeveniments que permeten capturar l'estat del procés quan són rebuts al control dels processos.

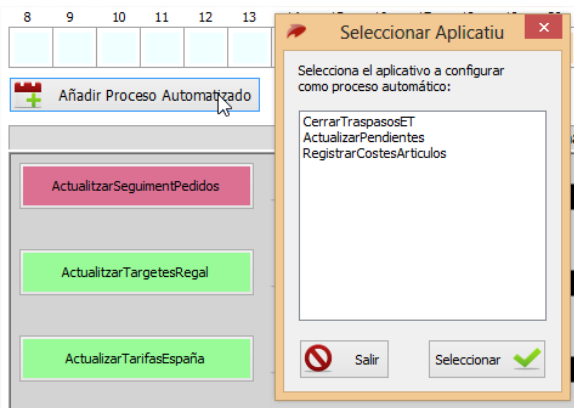
Tota aquesta informació visual i capacitat d'interacció està encapsulada en un **control d'usuari** <sup>33</sup>(*ctrlProcessos.vb*), i s'instancia de forma **dinàmica** cada cop que obrim el programa, basant-se en la corresponent taula del sistema (*processos*), que hem vist en la implementació de la base de dades en el capítol anterior.

Aquest control d'usuari també captura els events rebuts del Midgard, i els plasma sobre les barres de progrés i registre textual del control, fent que l'addició d'un nou procés en el futur, no impliqui cap cost en el disseny de la interfície. Simplement caldrà:

- Afegir-lo com a aplicació auto executable en la gestió d'aplicacions.
- Crear el codi associat al procés
- Crear el corresponent *mapping* en el gestor de processos automatitzats del SetelSysCore.
- Afegir el procés en la llista de processos actius.

<sup>33</sup> Creating a Windows Form User Control – [Microsoft MSDN](https://msdn.microsoft.com/en-us/library/aa479646.aspx)

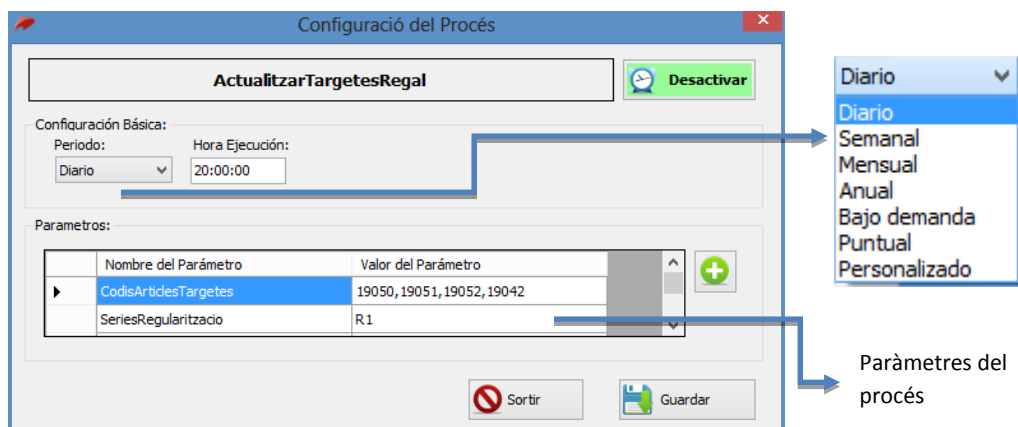
Per afegir un procés a la llista dels vigents, simplement cal fer un clic en el botó de “*Añadir Proceso Automatizado*” i ens apareixerà una finestra on podem seleccionar l’aplicatiu que volem afegir a la llista dels actuals.



En aquesta llista només ens apareixeran aplicacions que hem definit com a “automatitzables” en la gestió corresponent del sistema.

Finalment podem en tot moment accedir a la configuració del procés fent clic al botó del seu nom, per tal de poder configurar:

- Activar o desactivar-lo.
- Definir el tipus de programació
- Paràmetres necessaris per l’execució.





*Exemple de programació personalitzada (la més avançada i potent)*

The screenshot shows the 'Configuració del Procés' window for the process 'ActualizarTargetesRegal'. The window has a title bar with a red close button. Below the title bar is a section with a button 'ActualizarTargetesRegal' and a green 'Desactivar' button. The main configuration area is divided into several sections:

- Configuración Básica:** Includes 'Periodo:' set to 'Personalizado' and 'Patrón de ejecución:' set to 't1'. There are also buttons for adding, deleting, and editing patterns.
- Patrones de ejecución:** Includes 'Nombre patrón:' set to 'Test', 'Vigente desde:' set to '10/06/2014', and 'Vigente hasta:' set to '10/06/2014'. There is a checkbox 'Siempre está vigente' which is checked. To the right, there are fields for 'Hora:', 'Minuto:', and 'Segundo:' set to 5, 0, and 0 respectively. Below these are buttons for adding and deleting patterns.
- Horas de ejecución:** A table showing the days of the week and their corresponding execution times. The table has two columns: 'Dia' and 'Horas de ejecución'. The rows are:
 

Dia	Horas de ejecución
<input checked="" type="checkbox"/> Lunes	02:00:00 05:00:00
<input type="checkbox"/> Martes	
<input checked="" type="checkbox"/> Miércoles	
<input type="checkbox"/> Jueves	
<input checked="" type="checkbox"/> Viernes	
<input type="checkbox"/> Sábado	
<input type="checkbox"/> Domingo	
- Parametros:** A table with two columns: 'Nombre del Parámetro' and 'Valor del Parámetro'. The rows are:
 

Nombre del Parámetro	Valor del Parámetro
CodisArtidesTargetes	19050,19051,19052,19042
SeriesRegularitzacio	R1

At the bottom of the window, there are buttons for 'Cancelar', 'Guardar', 'Sortir', and 'Guardar'.

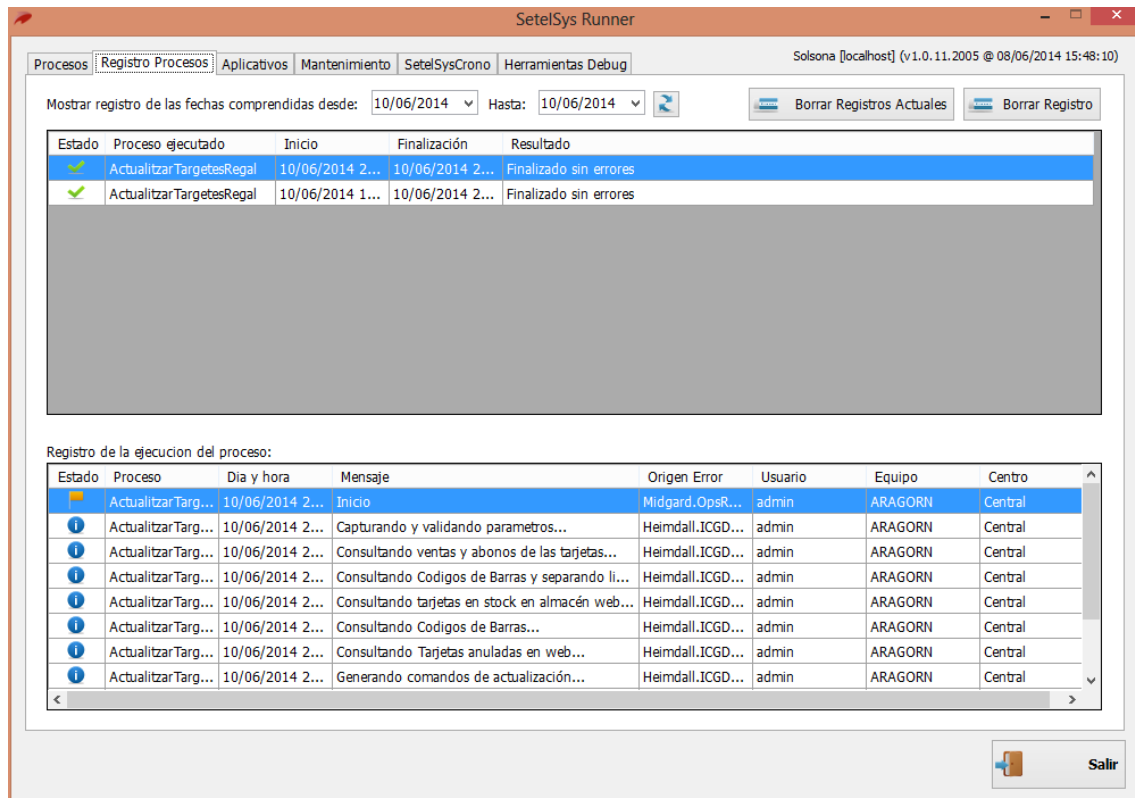
*Visualització del registre dels processos (Pestanya "Registro Procesos")*

Des d'aquesta pestanya, podem visualitzar el registre de cada execució d'un procés, podent visualitzar de cada execució:

- Estat de la finalització
- Procés executat
- Hora i dia d'inici.
- Hora i dia de finalització.
- Missatge textual identificador sobre el resultat.

De cada execució podem saber tot el detall de l'execució generat pels diferents events generats pel procés, podent-ne visualitzar:

- Estat de cada pas
- Nom del procés
- Missatge associat a cada pas.
- Origen de l'emissió de l'esdeveniment (objecte del sistema).
- Usuari i equip on s'han executat així com la seu.



També podem:

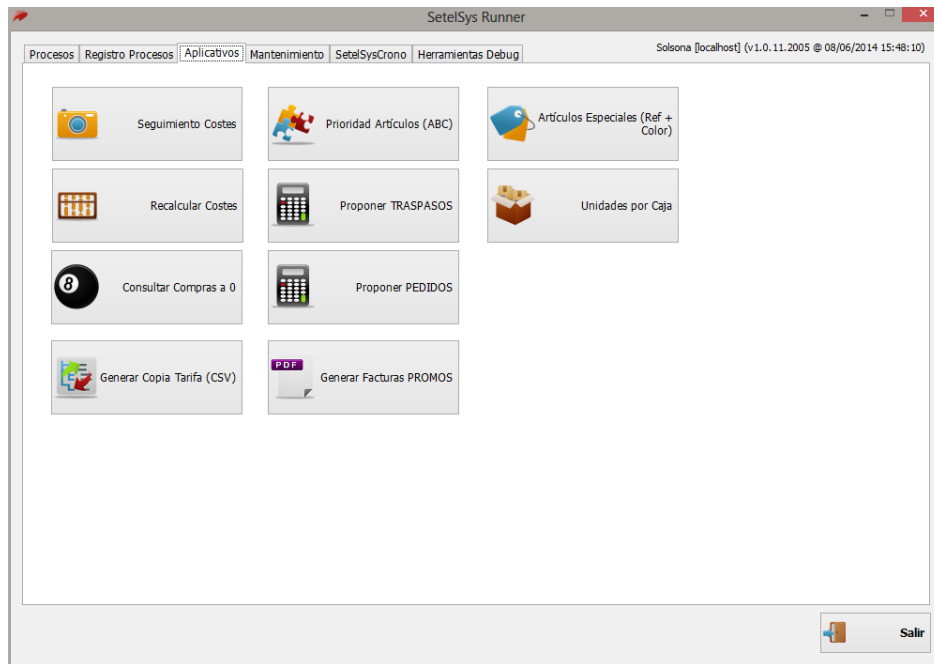
- Filtrar el registre d'activitat entre dos dates concretes.
- Eliminar tot el registre d'activitat
- Eliminar el registre d'activitat actualment visualitzat.

Tota aquesta informació s'extreu de la taula “logs\_processos” del sistema, on han sigut emmagatzemats pel **Midgard** al gestionar els esdeveniments i plasmar-los sobre la base de dades.

Recordem que en tot moment la comunicació amb les dades es fa a través del nucli del sistema **SetelSysCore**, respectant sempre els mètodes i funcions definits allà.

### *Aplicatius especialitzats (Pestanya “Aplicativos”)*

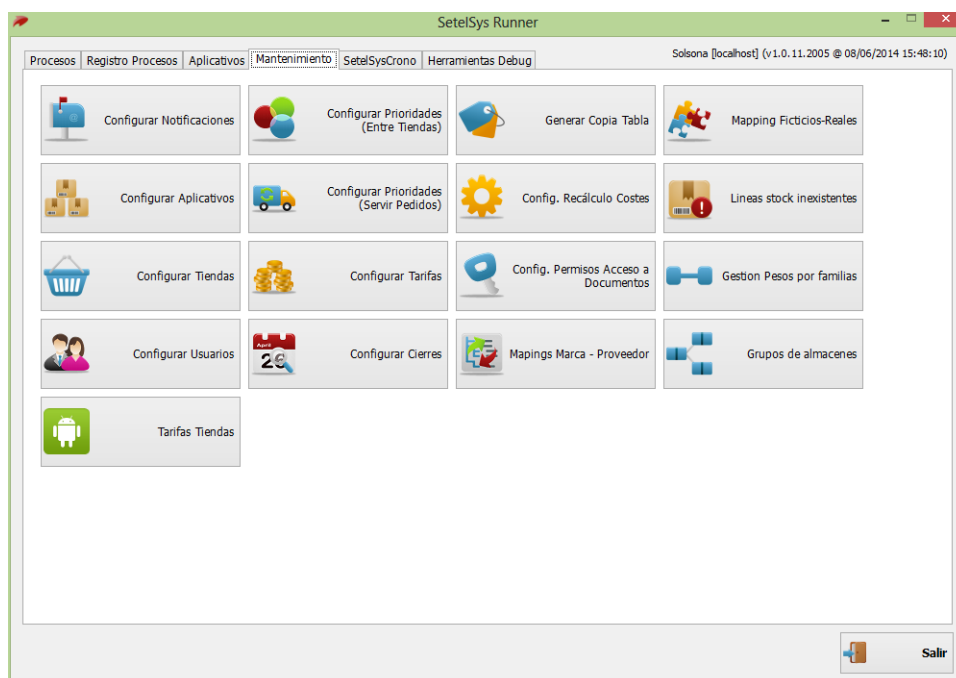
Com hem comentat anteriorment, s'han afegit dins del SetelSysRunner un ventall d'aplicacions destinades a operacions molt concretes sobre l'ERP i d'ús administratiu, de forma que no es volen posar accessibles dins de l'aplicació web del SetelSys.



No entrarem en detall en aquestes per ser centrades en resultats difícils d'interpretar sense tenir un coneixement avançat de la lògica de negoci de l'empresa, destinant més temps a la següent pestanya on considerem que les funcionalitats són de més interès.

### *Manteniment del sistema i parametrització d'aplicacions (Pestanya "Mantenimiento")*

En la següent pantalla podem trobar l'accés a la configuració de la major part d'elements del sistema, així com la parametrització de diferents aplicacions o parametritzacions que ho han requerit.



## Configurar notificaciones

Ens permet afegir diferents destinataris a qui s'enviarà un registre dels processos automatitzats de forma automatitzada cada dia a l'hora especificada. Podem afegir tants usuaris com es vulgui i es pot configurar tota la informació d'un servidor SMTP propi. Finalment, ens dona l'opció d'enviar un correu de notificació de forma manual.

Adjuntem a continuació una mostra on es veu un exemple del correu enviat:

### Registro Procesos Automáticos 08/06/14 07:00

SetelSys Runner

8 juny (fa 2 dies)



Resumen diario del registro de procesos automáticos

08/06/14 07:00

Estado	Proceso	Inicio	Fin	Mensaje
✓	RecalcuCostosUltimesCompres	06:48	06:48	Proceso finalizado correctamente.
✓	NetejarFTPEXchange	06:45	06:45	Proceso finalizado correctamente.
✓	CerrarPedidosNoStockET	06:30	06:30	Proceso finalizado correctamente.
✓	BorrarEtiquetas	06:05	06:05	Proceso finalizado correctamente.
✓	ActualizarTargetesRegal	06:00	06:01	Proceso finalizado correctamente.
✓	ActualizarTarifasEspaña	05:00	05:32	Proceso finalizado correctamente.
✓	ActualizarSeguimentPedidos	04:30	04:39	Proceso finalizado correctamente.
✓	RecalcuCostosUltimesCompres	03:30	03:30	Proceso finalizado correctamente.
✓	ActualizarSeguimentPedidos	16:00	16:08	Proceso finalizado correctamente.
✓	PedidosEntreTiendas	14:50	14:53	Proceso finalizado correctamente.

## Configurar Aplicatius

Des d'aquesta opció podem gestionar la creació, modificació i eliminació dels aplicatius existents en el sistema. Al crear un nou aplicatiu podem escollir el seu nom, la família a la qual pertany, el nom descriptiu i la descripció global. També podem anotar si el procés es pot **auto executar**, cosa que ens permetrà posteriorment poder-lo afegir en la llista de processos vigents.

Com veurem més endavant, la **família** anotada en l'aplicatiu també es important per tal de vincular-ho amb la secció corresponent d'aplicacions de l'aplicació web SetelSys.

Alta / Baja / Modificación de aplicativos

Nuevo aplicativo:

Nombre del aplicativo:

Familia de aplicativos:

Nombre descriptivo:

Descripción:

☐ Puede ejecutarse automáticamente?

Listado de aplicativos en SetelSys:

Nombre	Familia	Nombre descriptivo	Descripción	Puede Autoejecutar
ReservasPerBoti...	Logística	Reservas Para ...	Reservas seman...	<input type="checkbox"/>
PendientesServir	Logística	Pedidos Pendien...	Pedidos pendien...	<input type="checkbox"/>
FaltasStock	Logística	Faltas de Artículos	Faltas de artícul...	<input type="checkbox"/>
ActualizarTarge...	CRM	Actualizar Tarjet...	Proceso para ac...	<input checked="" type="checkbox"/>
ActualizarTarfas...	Tarifas	Actualizar Tarifa...	Proceso que gen...	<input checked="" type="checkbox"/>
MaterialPerDem...	Logística	Material Para Pe...	Listado agrupad...	<input type="checkbox"/>
ActualizarUbica...	Artículos	Actualizar Ubica...	Actualización de ...	<input type="checkbox"/>
StocksWebProv...	Artículos	Stocks Web Pro...	Aplicativo para g...	<input type="checkbox"/>
Minims	Artículos	Modificar Mínimo...	Modificación mas...	<input type="checkbox"/>
ArticulosBotiga	Artículos	Artículos En Tien...	Aplicativo para c...	<input type="checkbox"/>
Regularizacions	Artículos	Listado de Regul...	Listado de las re...	<input type="checkbox"/>
CrearArtides	Artículos	Crear Artículos e...	Creación de artí...	<input type="checkbox"/>
ActualizarAntici...	Otros	Actualizar Antici...	Aplicativos para ...	<input type="checkbox"/>
RestaurarTarifes	Tarifas	Restaurar Copia...	Restaurar copia...	<input type="checkbox"/>
CerrarTraspososET	Logística	Cerrar Traspaso...	Cerrar Traspaso...	<input checked="" type="checkbox"/>
CerrarPedidosN...	Logística	Cerrar Pedidos ...	Cerrar Pedidos ...	<input checked="" type="checkbox"/>
TraspassarVent...	Logística	Traspassar Venta...	Traspassar Venta...	<input type="checkbox"/>
TancarTraspasE...	Logística	Cerrar Traspaso...	Cerrar Traspaso...	<input type="checkbox"/>
NetejarFTPExch...	Mantenimiento	Limpiar FTP Exch...	Limpia los archiv...	<input checked="" type="checkbox"/>

## Configurar botigues

Des d'aquest formulari podem editar, eliminar o afegir noves **botigues** al sistema, especificant-ne e seu nom identificador, així com el mètode de detecció a utilitzar, la seva IP Pública i la subxarxa dins la xarxa privada virtual amb la que es connecten a la central (VPN).

També podem assignar **RDPs**<sup>34</sup> (arxius de configuració d'accés per Terminal Server) a la botiga, la qual cosa farà que estiguin disponibles per descarregar des de l'aplicació web de SetelSys.

La llista de botigues existents en el sistema, conformaran la **llista de botigues** que es podran seleccionar en el procés d'inici de sessió de l'usuari dins l'aplicació web de SetelSys.

<sup>34</sup> RDP: Remote Desktop Protocol – [Microsoft MSDN](https://docs.microsoft.com/en-us/windows-server/remote/remote-desktop-services/using-rdp)



Alta / Baja / Modificación de usuarios

Nombre Completo: Super Administrador, Barcelona, Madrid 2, Malaga, Solsona, Valencia 2, Zaragoza

Nombre usuario: admin, bcn, bcn2, mad, mad2, mal, sol, val, val2, zar

Datos del usuario:

Nombre de usuario:

Nombre Completo:

Estado:

Contraseña:

Contraseña (reescribir):

RDPs: ☐

Tiendas autorizadas:

Máquinas autorizadas:

Aplicativos autorizados:

Desahacer, Guardar, Salir

Botigues autoritzades      Màquines autoritzades      Aplicatius autoritzats

### Configurar prioritats entre botigues

Aquest formulari s'utilitza per a parametritzar les prioritats existents entre els diferents magatzems de les botigues, en el moment en que s'ha de decidir de quin extreure-ho. La prioritat s'estableix a través d'una cadena numèrica:

- 0 → La prioritat més alta
- 12 → La prioritat més baixa

Fent clic amb el botó dret o esquerre sobre una cel·la combinada per un magatzem d'origen i un de destí, s'incrementa o decrementa respectivament l'ordre de prioritats de la cel·la.

Un exemple: *El magatzem E11 requereix material, i d'aquest material ni ha a 3 magatzems: E14, E18, E12. Per ordre de prioritats tenim que el pes de E14 és 6, el de E18 és 5 i el de E12 és 7. El pes menor és E18 i per tant farem venir el material d'aquest magatzem.*

Configuración de prioridades entre almacenes

Almacenes que SIRVEN MATERIAL													
	A11	A12	A13	E11	E12	E13	E14	E16	E17	E18	E19	E1D	E1Z
E11	10	11	12	0	7	1	6	4	3	5	2	9	8
E12	10	11	12	7	0	1	5	4	3	2	6	9	8
E14	10	11	12	8	7	1	0	3	2	4	5	9	6
E15	8	10	11	6	12	0	4	3	2	1	5	9	7
E16	10	11	12	8	7	1	3	0	2	4	5	9	6
E17	10	11	12	8	7	1	2	3	0	4	5	9	6
E18	10	11	12	8	7	1	4	3	2	0	5	9	6
E19	10	11	12	8	7	1	5	3	2	4	0	9	6
E1D	10	11	12	8	7	1	6	2	3	4	5	0	9
E1Z	10	11	12	8	7	1	6	3	2	4	5	9	0

Desahacer, Guardar, Salir

### Configurar prioritats per al “Servir Pedidos”

Igual que en el cas anterior, aquest formulari s'utilitza per parametritzar una aplicació. En concret pel procés de servir comandes que s'utilitza per a enviar material directament al client final des del magatzem central o una botiga on hi ha existències de tota la comanda. En el cas d'haver-se d'enviar des d'una botiga, hem de poder establir prioritats per cada zona geogràfica que pot arribar a establir-se a nivell de:

- País
- Comunitat
- Província
- Municipi

Igual que abans, utilitzem una columna numèrica per definir l'ordre de prioritats per a cada entitat geogràfica. Fent doble clic sobre els títols de les entitats entrem dins el nivell següent, i fent un clic esquerre tornem a l'anterior.

	E11	E12	E13	E14	E16	E17	E18	E19	E10	E12
ANDALUCIA	5	7	0	3	1	2	6	4		8
ANDORRA	2	7	0	5	3	4	6	1		8
ARAGON	4	2	0	6	7	5	1	3		8
BALEARES	4	2	0	6	7	5	1	3		8
CANARIAS	5	7	0	3	1	2	6	4		8
CANTABRIA	2	4	0	6	7	5	3	1		8
CASTILLA Y LEON	2	4	0	6	7	5	3	1		8
CASTILLA-LA MANCHA	4	2	0	6	7	5	1	3		8
CATALUÑA	4	2	0	6	7	5	1	3		8
CEUTA	5	7	0	3	1	2	6	4		8
COMUNIDAD DE MADRID	2	4	0	6	7	5	3	1		8
COMUNIDAD VALENCIANA	4	6	0	2	7	1	5	3		8
EXTREMADURA	4	2	0	6	7	5	1	3		8
GALICIA	2	4	0	6	7	5	3	1		8
LA RIOJA	4	2	0	6	7	5	1	3		8
MELILLA	5	7	0	3	1	2	6	4		8
NAVARRA	4	2	0	6	7	5	1	3		8
PAIS VASCO	4	2	0	6	7	5	1	3		8

### Configurar tarifas

Des d'aquest formulari especifiquem quines tarifes s'hauran de tornar a calcular o regenerar durant el procés automàtic “Actualizar Tarifas”.

### I podríem continuar...

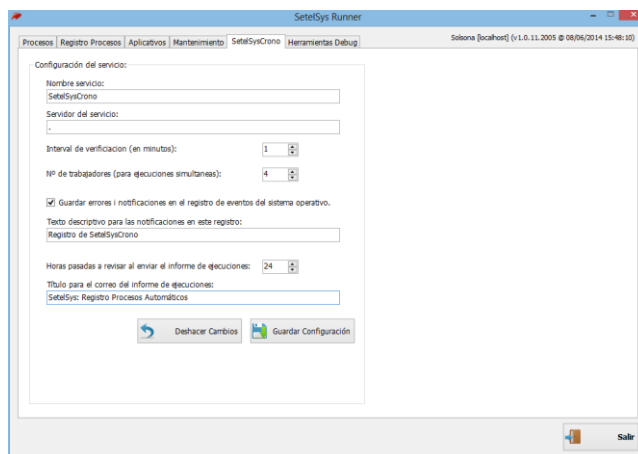
Podríem continuar analitzant la resta de formularis de parametrització, però no veiem convenient ni interessant insistir-hi, tot i que tots els aplicatius són 100% funcionals actualment. Degut a que per la correcta interpretació de l'objectiu de les parametritzacions cal explicar en profunditat el model de negoci associat, deixem d'introduir-nos més profundament en la resta d'opcions d'aquesta pestanya.



### *Configuracions del servei de Windows SetelSysWinService (Pestanya "SetelSysCrono")*

En aquest formulari podem configurar certs valors referents al servei de Windows SetelSysWinService que utilitza la classe SetelSysCrono per l'execució de processos automatitzats. En concret els valors que podem editar:

- Nom del servei
- Servidor que hostatja el servei (per defecte "." = localhost).
- Interval de verificació de processos pendents d'executar.
- Nº de treballadors assignats al servei (*BackgroundWorkers*)
- Activar / Desactivar si volem que els errors del servei es registrin en el registre d'events propi de Windows.
- Títol descriptiu per a les notificacions en el registre de Windows.
- Títol per al correu d'enviament de notificacions.



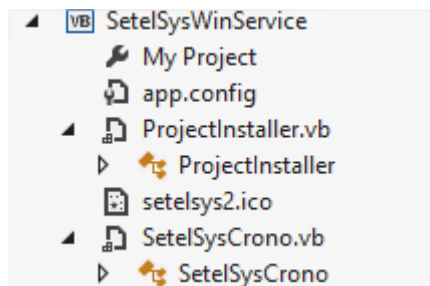
### *Eines de Debug (Pestanya "Herramientas Debug")*

En aquesta pestanya tenim accés a diferents eines internes de verificació. Només es pot activar a través de la combinació de tecles "Ctrl+Alt+D".

### 7.6.3 SetelSysWinService

Aquest component serà l'encarregat d'implementar un servei de Windows<sup>35</sup> per verificar de forma intermitent les necessitats d'execució d'algun dels processos automatitzats del sistema.

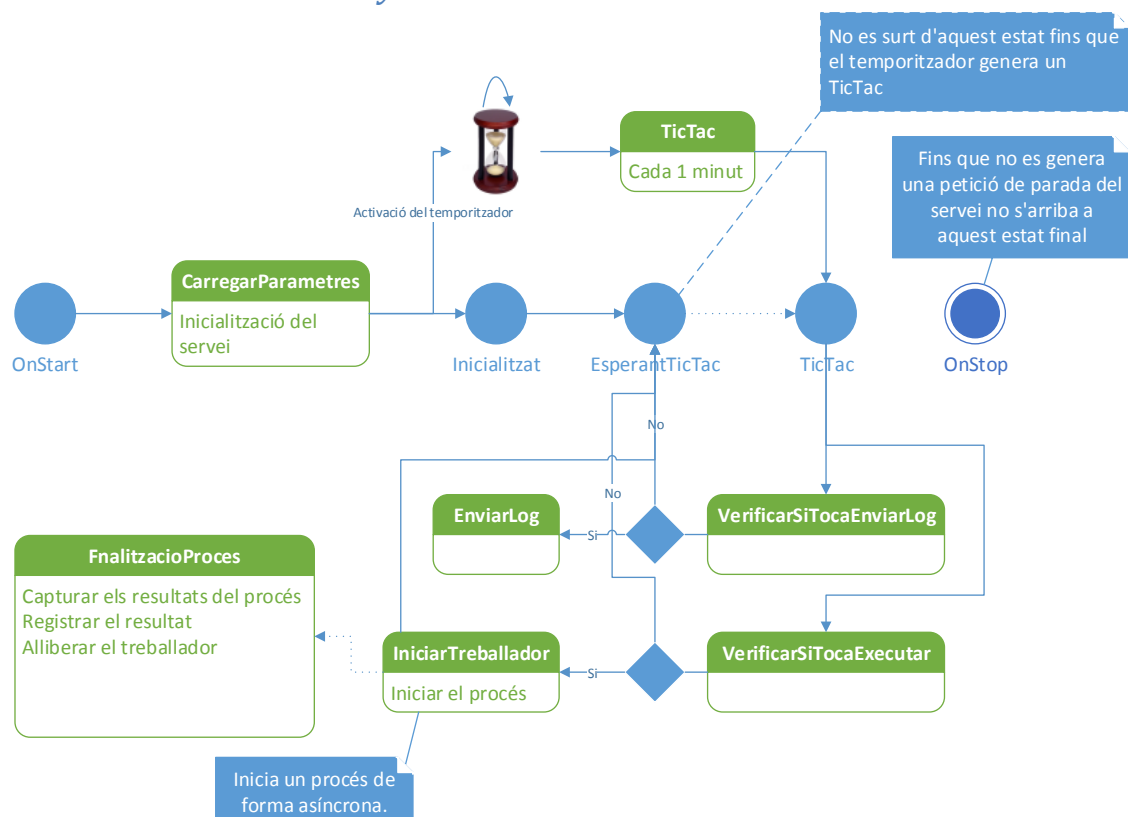
Veiem tot seguit una figura indicativa dels diferents elements que componen el projecte:



Cal destacar-ne:

- **ProjectInstaller**: Classe encarregada de gestionar la instal·lació del servei, en la qual es defineixen totes les accions necessàries a executar en els diferents esdeveniments del sistema, durant el procés d'instal·lació del servei.
- **SetelSysCrono**: Classe encarregada de la gestió de la funcionalitat del servei, consistent en verificar

#### Cicle de vida del servei SetelSysCrono:



<sup>35</sup> Create Windows Services – [Microsoft MSDN](https://docs.microsoft.com/en-us/windows/desktop/svc/create-windows-services)

Com es pot observar en el diagrama, cal destacar diferents punts:

- L'existència d'un temporitzador que s'activa cada 60 segons, per iniciar el procés de verificació de les tasques a fer.
- Les tasques a verificar en cada activació del temporitzador són:
  - Verificar si toca enviar el registre de processos (1 cop al dia)
  - Verificar si toca executar algun procés automatitzat (depenent de la planificació dels processos).
- En cas de que s'executi un procés automatitzat, aquest es fa de forma asíncrona, deixant el fil principal del servei disponible a la gestió de nous esdeveniments del temporitzador i enllaçant la tasca pendent a un treballador en un fil independent.

### *Sobrecàrrega del servei*

El servei està preparat per acceptar l'execució simultània de fins a **10** processos declarats en una llista de treballadors.

- En cada execució d'un procés es cerca el primer treballador que no estigui ocupat i se li assigna la tasca.
- En cas de que tots els treballadors estiguin ocupats el servei emet una excepció i posposa l'execució fins a que existeixi un treballador lliure.

Per tant el servei es capaç d'acceptar una alta sobrecàrrega de processos executats de forma automàtica o que es solapin en el temps. Tot i que ara mateix no es requereixen processos automatitzats i és millor no fer-ho per no abusar de la disponibilitat de la base de dades o detectar possibles problemes amb la concurrència d'accés a dades, el servei està preparat per que en un futur es pugui treballar concurrentment amb diferents processos automatitzats a través de **Multi Threading** utilitzant objectes del tipus *BackgroundWorker*<sup>36</sup>.

### *Implementació de la classe SetelSysCrono*

Tot seguit mostrem la definició de la classe implementada per SetelSysCrono, deixant únicament la definició d'atributs privats i les capçaleres de mètodes i funcions:

Public Class SetelSysCrono

```
Private _PROCS As List(Of InfoProces)
Private _M As Midgard
Private _CLI As Midgard.InfoClient
Private WithEvents _MAILER As Mercury.EnviamentLogsPerCorreu
Private WithEvents _TIMER As Timers.Timer
Private WithEvents _LOG As EventLog

Private _TEMPS_REFRESC As Integer
```

<sup>36</sup> BackgroundWorker Class (.NET Framework 4.5) – [Microsoft MSDN](#)

```

Private _CONF As ConfiguracioSetelSys

Private _WORKERS() As System.ComponentModel.BackgroundWorker
Private _MIDGARDS() As Midgard
Private _SRV_NAME As String
Private _SRV_LOG_DESCRIPTION As String
Private _SEU As Midgard.CentralMTC

Protected Overrides Sub OnStart(ByVal args() As String)
Protected Overrides Sub OnStop()

Private Sub CarregarParametres()

Private Sub TicTac(sender As Object, e As Timers.ElapsedEventArgs) Handles _TIMER.Elapsed

Private Sub VerificarSiTocaEnviarLog(Ara As DateTime)
Private Sub VerificarSiTocaExecutar(Ara As DateTime)

Private Sub IniciarTreballador(Peticio As Midgard.InfoIniciExecucio)
Private Sub FinalitzacioProces(Resultat As Midgard.ResultatExecucio)

Private Sub T(Ex As Exception)

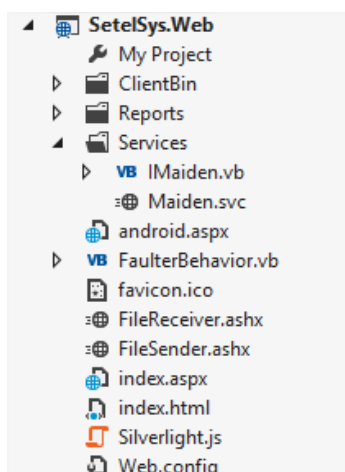
End Class

```

#### 7.6.4 Maiden: Servei web

Dins del projecte **SetelSys.Web** hem definit el servei web Maiden, basat en l'arquitectura de serveis WCF <sup>37</sup>de Microsoft.

Aquest servei serà l'encarregat de transmetre totes les peticions de l'aplicatiu Silverlight al nucli del sistema resident en el servidor d'aplicacions i dades, donant accés a tota la lògica de negoci necessària per mostrar a l'usuari final.



<sup>37</sup> Hosting and consuming a WCF Service – [Microsoft MSDN](https://msdn.microsoft.com/en-us/library/ff649612.aspx)

El servei consta de 2 arxius:

- **IMaiden.vb**: Definició de la interfície d'operacions necessàries per consultar sobre el nucli.
- **Maiden.svc**: Implementació de l'anterior interfície, on s'associa el codi final a cada operació definida en la interfície.

En cada operació implementada en el Maiden, cal instanciar una connexió amb el nucli a través del Midgard, a través del qual accedirem directament a les funcions integrades en el gestor d'operacions corresponent, com es podrà observar en el codi annex en la part inferior.

### *Definició dels paràmetres de servei en l'arxiu de configuració de la web (web.config)*

Tot seguit mostrem el contingut de l'arxiu **web.config**<sup>38</sup> encarregat de gestionar la definició dels paràmetres del servei, tant a nivell d'enllaç com tractament d'errors:

```
<system.serviceModel>
  <extensions>
    <behaviorExtensions>
      <add name="Faulter" type="SetelSys.Web.FaulterBehavior, SetelSys.Web, Version=1.0.0.0, Culture=neutral,
PublicKeyToken=null"/>
    </behaviorExtensions>
  </extensions>

  <behaviors>
    <serviceBehaviors>
      <behavior name="MaidenBehavior">
        <serviceMetadata httpGetEnabled="true" />
        <serviceDebug includeExceptionDetailInFaults="true" />
        <dataContractSerializer maxItemsInObjectGraph="2147483647" />
      </behavior>
    </serviceBehaviors>
  </behaviors>

  <services>
    <service name="SetelSys.Web.Maiden" behaviorConfiguration="MaidenBehavior">
      <endpoint address="" binding="basicHttpBinding" bindingConfiguration="MaidenBinding" contract="SetelSys.Web.IMaiden"
behaviorConfiguration="Faulter" />
    </service>
  </services>

  <serviceHostingEnvironment aspNetCompatibilityEnabled="true" multipleSiteBindingsEnabled="true" />

  <bindings>
    <basicHttpBinding>
      <binding name="MaidenBinding" hostNameComparisonMode="StrongWildcard" receiveTimeout="10:00:00"
sendTimeout="10:00:00" openTimeout="10:00:00" closeTimeout="10:00:00" maxReceivedMessageSize="2147483647"
maxBufferSize="2147483647" maxBufferPoolSize="2147483647" transferMode="Buffered" messageEncoding="Text"
textEncoding="utf-8" bypassProxyOnLocal="false" useDefaultWebProxy="true">
        <security mode="None" />
      </binding>
    </basicHttpBinding>
  </bindings>
</system.serviceModel>
```

<sup>38</sup> ASP.NET Web Configuration Guidelines – [Microsoft MSDN](#)

## Comunicació d'excepcions a través del servei

En el cas de que es generi algun tipus d'excepció, hem de transmetre-la a l'extrem consumidor del servei perquè rebi la informació necessària sobre l'error. Com s'ha vist en l'anterior codi de l'arxiu web.config:

```
<behaviorExtensions>
  <add name="Faulter" type="SetelSys.Web.FaulterBehavior, SetelSys.Web, Version=1.0.0.0, Culture=neutral,
PublicKeyToken=null"/>
</behaviorExtensions>
.....
.....
<service name="SetelSys.Web.Maiden" behaviorConfiguration="MaidenBehavior">
  <endpoint address="" binding="basicHttpBinding" bindingConfiguration="MaidenBinding" contract="SetelSys.Web.IMaiden"
behaviorConfiguration="Faulter" />
</service>
```

Estem declarant un comportament d'excepcions que assignem posteriorment a la definició del servei perquè es puguin transmetre. En concret l'encarregat de definir el comportament serà la classe **FaulterBehavior**:

```
Public Class FaulterBehavior
  Inherits System.ServiceModel.Configuration.BehaviorExtensionElement
  Implements System.ServiceModel.Description.IEndpointBehavior

  Public Sub ApplyDispatchBehavior(endpoint As System.ServiceModel.Description.ServiceEndpoint, endpointDispatcher As
System.ServiceModel.Dispatcher.EndpointDispatcher) Implements
System.ServiceModel.Description.IEndpointBehavior.ApplyDispatchBehavior

  Public Class FaulterInspector
    Implements System.ServiceModel.Dispatcher.IDispatchMessageInspector

    Public Function AfterReceiveRequest(ByRef request As System.ServiceModel.Channels.Message, channel As
System.ServiceModel.IClientChannel, instanceContext As System.ServiceModel.InstanceContext) As Object Implements
System.ServiceModel.Dispatcher.IDispatchMessageInspector.AfterReceiveRequest

    Public Sub BeforeSendReply(ByRef reply As System.ServiceModel.Channels.Message, correlationState As Object) Implements
System.ServiceModel.Dispatcher.IDispatchMessageInspector.BeforeSendReply

  End Class

  Public Overrides ReadOnly Property BehaviorType As System.Type
  Protected Overrides Function CreateBehavior() As Object
  Public Sub Validate(endpoint As System.ServiceModel.Description.ServiceEndpoint) Implements
  Public Sub AddBindingParameters(endpoint As System.ServiceModel.Description.ServiceEndpoint, bindingParameters As
  Public Sub ApplyClientBehavior(endpoint As System.ServiceModel.Description.ServiceEndpoint, clientRuntime As

End Class
```

Finalment, i com veurem més endavant, al detectar qualsevol excepció en la implementació de la interfície, caldrà llançar una excepció del tipus **FaultException**:

```
Private Function T(ex As Exception) As FaultException(Of ErrorSistema)
  If IsNothing(ex) Then Return New FaultException("Excepció nula", New FaultCode("NullEx"), "T(ex)")
  Return New FaultException(Of ErrorSistema)(ParseException(ex))
End Function
```

### Estructura del fitxer IMaiden.vb

Hem ajustat el codi el màxim possible perquè sigui llegible.

```
<ServiceContract()>
Public Interface IMaiden

    '===== OPERACIONS BÀSIQUES D'AUTENTICACIÓ I CONFIGURACIÓ INICIAL =====
    <OperationContract()> <FaultContract(GetType(ErrorSistema))>
    Function GetClientInfo(Client As InfoClient) As Maiden.ConexioRemota

    <OperationContract()> <FaultContract(GetType(ErrorSistema))>
    Function GetNomsBotigues(Client As InfoClient) As List(Of Botiga)

    <OperationContract()> <FaultContract(GetType(ErrorSistema))>
    Function ValidarUsuari(Client As InfoClient, Usuari As String, Password As String, NomCurtBotiga As String, Maquina As String)
        As SetelSysDataCom.ResultatValidacioUsuari

    <OperationContract()> <FaultContract(GetType(ErrorSistema))>
    Function EsUsuariGestorConsultes(Client As InfoClient, NomUsuari As String) As Boolean

    <OperationContract()> <FaultContract(GetType(ErrorSistema))>
    Function GetAuthOps(Client As InfoClient) As List(Of SetelSysDataCom.Aplicatiu)

    '===== IMPRESIO REMOTA =====
    <OperationContract()> <FaultContract(GetType(ErrorSistema))>
    Function GenerarReportArticlesCreates(Client As InfoClient, Format As Seshat.FormatSortida, Tipus As Seshat.TipusReport,
        Dades As List(Of repLiniaArticlesCreates)) As String

    <OperationContract()> <FaultContract(GetType(ErrorSistema))>
    Function GenerarReportMaterialPerRebre(Client As InfoClient, Format As Seshat.FormatSortida, BotigaDesti As String,
        Pedidos As List(Of PedidoPendent), PerRebre As List(Of InfoMaterialPerReservar)) As
        String

    <OperationContract()> <FaultContract(GetType(ErrorSistema))>
    Function GenerarInformeAmpliatPendantsServir(Client As InfoClient, Format As Seshat.FormatSortida, NomCurtBotiga As String,
        Pendants As List(Of PedidoPendent), IncloureSenseStock As Boolean, BuscarAnticipos As Boolean,
        BuscarCompres As Boolean)

    <OperationContract()> <FaultContract(GetType(ErrorSistema))>
    Function GenerarInformeBasicPendantsServir(Client As InfoClient, Format As Seshat.FormatSortida, NomCurtBotiga As String,
        Pendants As List(Of PedidoPendent), ImprimirConsultes As Boolean)

    ...
    Rest a d'operacions
    ...

End Interface
```

### Estructura del fitxer Maiden.svc

Hem ajustat el codi el màxim possible perquè sigui llegible.

```
Public Class Maiden
    Implements IMaiden

    Public Class ConexioRemota
        Public Property IPPublica As String
        Public Property IPPrivada As String
        Public Property Maquina As String
        Public Property Versio As String
    End Class

    #Region "===== OPERACIONS BÀSIQUES D'AUTENTICACIÓ I CONFIGURACIÓ INICIAL ====="
```

```

Public Function ValidarUsuari(Client As InfoClient, Usuari As String, Password As String, NomCurtBotiga As String, Maquina As String) As SetelSysDataCom.ResultatValidacioUsuari Implements IMaiden.ValidarUsuari
    Try
        Dim M As New SetelSysCore.Midgard(Client, CentralBotiga(NomCurtBotiga))
        Return M.SetelSys.Usuaris.ValidarUsuari(Usuari, Password, NomCurtBotiga, Maquina)
    Catch ex As Exception
        Throw T(ex)
    End Try
End Function

Public Function EsUsuariGestorConsultes(Client As InfoClient, NomUsuari As String) As Boolean Implements IMaiden.EsUsuariGestorConsultes
    Try
        Dim M As New SetelSysCore.Midgard(Client, CentralBotiga(Client.NomBotiga))
        Return M.SetelSys.Usuaris.EsUsuariGestorConsultes(NomUsuari)
    Catch ex As Exception
        Throw T(ex)
    End Try
End Function

Public Function GetNomsBotigues(Client As InfoClient) As List(Of Botiga) Implements IMaiden.GetNomsBotigues
    Try
        Dim M As New SetelSysCore.Midgard(Client, CentralBotiga(Client.NomBotiga) )
        Return M.SetelSys.Botigues.ConsultarTotesBotigues
    Catch ex As Exception
        Throw T(ex)
    End Try
End Function

Public Function GetClientInfo(Client As InfoClient) As ConexioRemota Implements IMaiden.GetClientInfo
    Try
        Dim Con As New ConexioRemota
        Con.Maquina = System.Web.HttpContext.Current.Request.UserHostName
        Con.IPPrivada = System.Web.HttpContext.Current.Request.UserHostAddress
        Con.IPPublica = System.Web.HttpContext.Current.Request.UserHostAddress
        Con.Versio = SetelSysCore.Midgard._VERSIO
        Return Con
    Catch ex As Exception
        Throw T(ex)
    End Try
End Function

Public Function GetAuthOps(Client As InfoClient) As List(Of SetelSysDataCom.Aplicatiu) Implements IMaiden.GetAuthOps
    Try
        Dim M As New Midgard(Client, CentralBotiga(Client.NomBotiga))
        Return M.SetelSys.Usuaris.ConsultarAplicatiuAmbPermisUsuari(Client.Usuari)
    Catch ex As Exception
        Throw T(ex)
    End Try
End Function
#End Region

...
Rest a d'operacions
...

Private Function T(ex As Exception) As FaultException(Of ErrorSistema)
    If IsNothing(ex) Then Return New FaultException("Excepcion nula", New FaultCode("NullEx"), "T(ex)")
    Return New FaultException(Of ErrorSistema)(ParseException(ex))
End Function

End Class

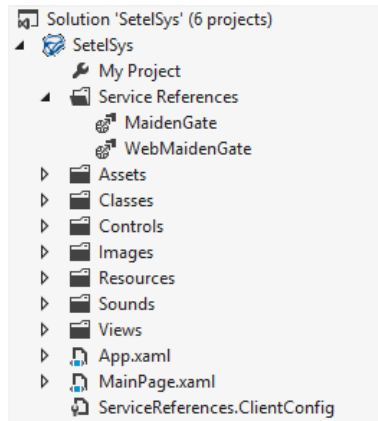
```



### 7.6.5 SetelSys: Aplicació web amb Silverlight.

Aquesta aplicació és l'encarregada d'implementar la interfície d'usuari a través d'una aplicació web encapsulada amb Silverlight.

#### *Estructura del projecte .NET:*



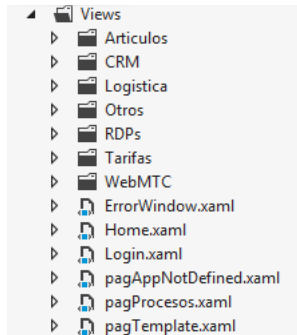
Expliquem a continuació cada una de les carpetes i elements principals que componen el projecte d'implementació de SetelSys amb Silverlight<sup>39</sup>:

- **Service References + ServiceReferences.ClientConfig:** Defineixen les referències i dades d'enllaç dels serveis web Maiden i WebMaiden, per tal de poder accedir a les funcionalitats del sistema. Fan referència doncs als serveis vistos en el punt anterior.
- **Assets:** Carpeta on es defineixen tots els arxius de tipus XAML que contenen les definicions d'estils, tipus de color, pinzells, fonts,... Són arxius que defineixen el comportament visual de l'estil de l'aplicació.
- **Classes:** Classes d'ús intern per diferents tasques en les diferents pàgines existents.
- **Controls:** Definició de finestres i controls d'ús comú al llarg de l'aplicació per poder-los reutilitzar en tots els punts possibles.
- **Images:** Arxius de tipus PNG, JPG o GIF que utilitzem en diferents controls de la interfície (botons, graelles,...)
- **Resources:** Arxius de definició de recursos associats a l'idioma. En aquest cas la interfície s'ha desenvolupat únicament en castellà, però si es volgués associar un nou idioma, caldria especificar el corresponent arxiu de recursos en aquesta carpeta i enllaçar-lo degudament en l'aplicació.
- **Sounds:** Arxius de so. Actualment no té cap contingut.
- **Views:** Aquí és on resideixen les diferents pàgines i finestres que es mostren en tota la interfície. Veurem més detalladament el seu contingut més endavant.
- **App.xaml:** Definició d'esdeveniments i mètodes especials a nivell de tota l'aplicació. És també l'encarregat d'inicialitzar tota l'aplicació en temps d'execució.
- **MainPage.xaml:** Pàgina principal mare, que conté l'estructura de la finestra principal, en la qual en un contenidor de tipus FRAME es van carregant les diferents pàgines.

<sup>39</sup> Create a new Silverlight Project – [Microsoft MSDN](#)

### *Vistes existents en el projecte (Carpeta “Views”):*

En la següent imatge es pot apreciar el contingut intern de la carpeta “Views”, contenidora de totes les pàgines i finestres del projecte:

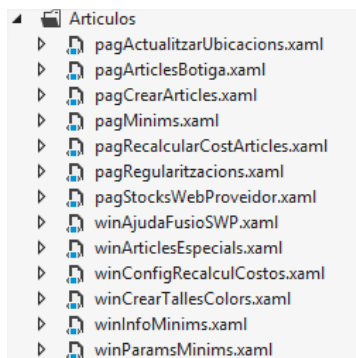


S’ha definit una subcarpeta per a cada **família d’aplicacions**, dins la qual resideixen totes les pàgines i finestres associades únicament a aquesta família.

A part existeixen un seguit de finestres i pàgines genèriques a l’arrel de la carpeta de vistes:

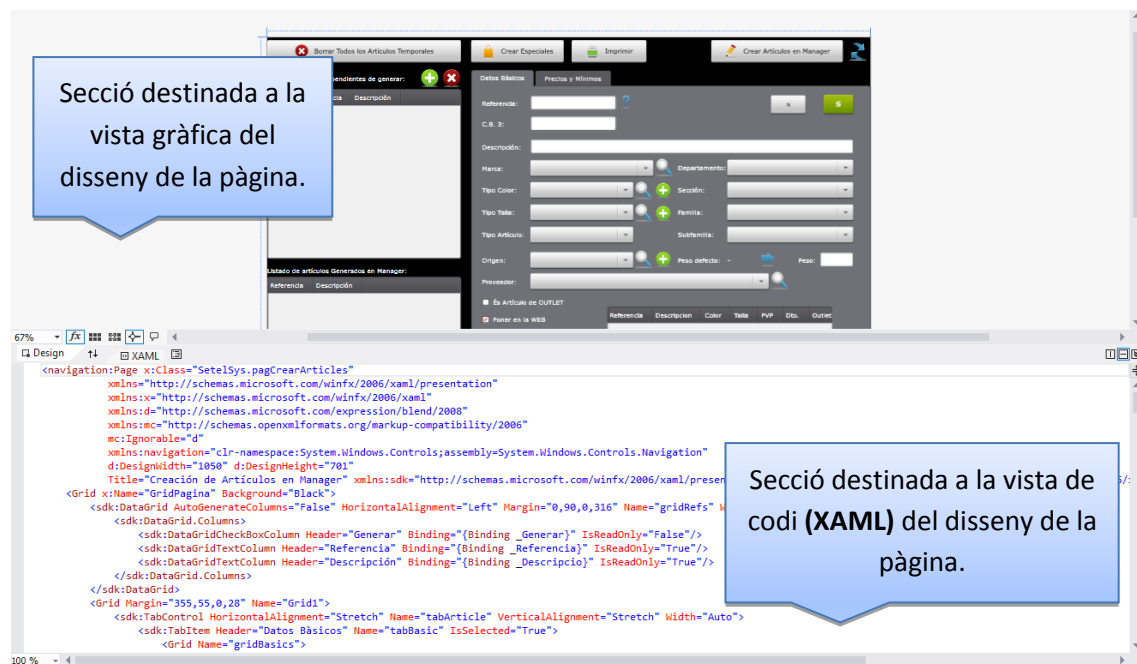
- *ErrorWindow.xaml*: Destinada a mostrar una finestra d’error per possibles excepcions no controlades per codi.
- *Home.xaml*: Pàgina inicial de càrrega a l’inici del sistema durant la pantalla d’inici de sessió.
- *Login.xaml*: Finestra encarregada de gestionar l’inici de sessió de l’usuari.
- *pagAppNotDefined.xaml*: Pàgina que es mostra quan s’està intentant carregar una pàgina que no es troba en el sistema.
- *pagProcesos.xaml*: Pàgina encarregada de mostrar la informació específica en l’execució de processos automatitzats de forma diferida.
- *pagTemplate.xaml*: Pàgina base que hem utilitzat per instanciar noves pàgines, al contenir la configuració i format bàsic que volem donar a totes les de l’aplicació.

Veient un exemple de les carpetes que hem vist, dins la carpeta “Articulos” podem trobar els següents fitxers:



Per conveni, tots els elements que són heretats de la classe “Page” del Silverlight, s’han anomenat amb les lletres “pag” com a prefix. De la mateixa manera, totes les finestres que hereden de la classe “ChildWindow” de Silverlight, s’han anomenat amb el prefix “win”.

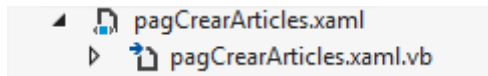
Cada un d’aquests elements és una pàgina o finestra que es mostra durant l’aplicació. Per exemple, veiem l’estructura des de la pantalla de desenvolupament, de la pàgina “pagCrearArticles.xaml”.



De la mateixa manera podem veure un exemple de la finestra “winCrearTallesColors.xaml”:



Totes les pàgines i finestres disposen d'una interfície gràfica (editable en vista disseny o vista de codi XAML<sup>40</sup>) i un controlador per codi assignat directament a dita pàgina o finestra:



En la imatge anterior podem veure com dins de la definició gràfica de la classe *pagCrearArticles.xaml* existeix el codi associat a través de *pagCrearArticles.xaml.vb*. Si entrem dins d'aquesta classe podem observar el codi assignat al control de tots els esdeveniment i lògica associada a la pàgina:

Partial Public Class *pagCrearArticles* Inherits Page

```
Public Sub New()
    Initialize()
    ....
```

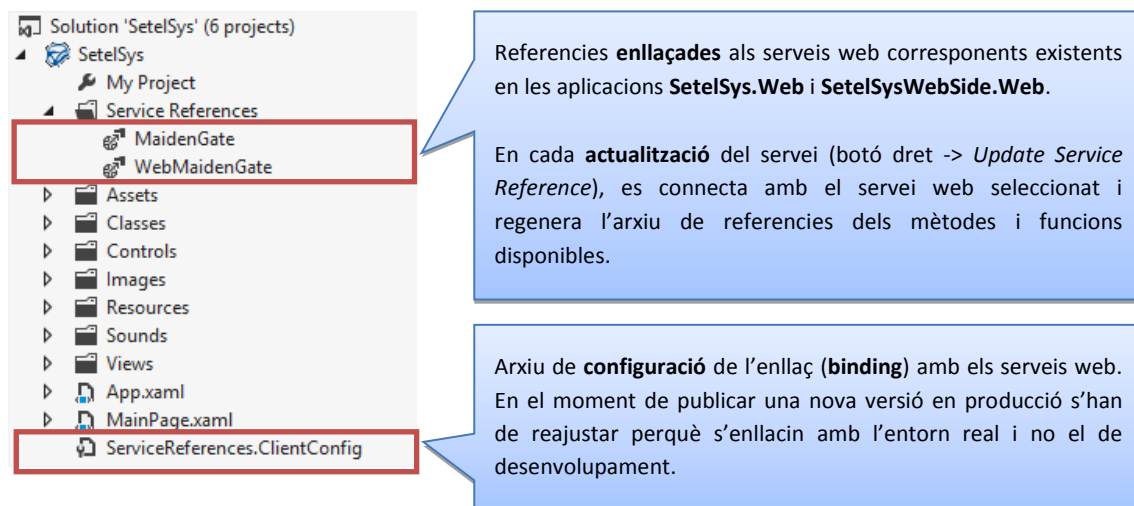
```
End Sub
```

```
...
```

End Class

### Enllaç de SetelSys amb els serveis web Maiden i WebMaiden<sup>41</sup>

Veiem en la següent imatge els elements que entren en joc en l'enllaç dels serveis web:



Veiem tot seguit l'arxiu de configuració dels serveis: *ServiceReferences.ClientConfig*, on s'especifica la configuració de l'enllaç dels clients cap als serveis.

<sup>40</sup> What is XAML? – [Microsoft MSDN](#)

<sup>41</sup> Using WCF Service with Silverlight – [Code Project](#)

```
<configuration>
  <system.serviceModel>

    <bindings>
      <basicHttpBinding>
        <binding name="BasicHttpBinding_IMaiden" receiveTimeout="10:00:00"
          sendTimeout="10:00:00" maxBufferSize="2147483647" maxReceivedMessageSize="2147483647">
          <security mode="None" />
        </binding>
        <binding name="BasicHttpBinding_IWebMaiden" maxBufferSize="2147483647"
          maxReceivedMessageSize="2147483647">
          <security mode="None" />
        </binding>
      </basicHttpBinding>
    </bindings>

    <client>
      <endpoint address="http://localhost/SetelSys/Services/Maiden.svc"
        binding="basicHttpBinding" bindingConfiguration="BasicHttpBinding_IMaiden"
        contract="MaidenGate.IMaiden" name="BasicHttpBinding_IMaiden" />
      <endpoint address="http://localhost/SetelSysWebSide/WebMaiden.svc"
        binding="basicHttpBinding" bindingConfiguration="BasicHttpBinding_IWebMaiden"
        contract="WebMaidenGate.IWebMaiden" name="BasicHttpBinding_IWebMaiden" />
    </client>

  </system.serviceModel>
</configuration>
```

En el primer bloc `<bindings>` definim la tipologia de l'enllaç per a cada servei, com poden ser els temps de *timeout*, i les capacitats dels *buffers*.

En el segon bloc `<client>` definim les adreces de consum dels serveis així com l'enllaç que han d'utilitzar definit en el primer bloc.

### Consum dels serveis web

En el moment que volem consumir algun dels mètodes o funcions disponibles en els serveis, caldrà seguir els següents passos:

- 1) Instanciar una connexió amb el servei web, assignant el tipus d'enllaç i la direcció del servei.
- 2) Assignar la rutina encarregada de tractar el retorn (*callback*) de l'execució del servei utilitzant el mètode (*addhandler*), o bé definir la connexió amb el servei amb events (*with events*) per posteriorment definir una rutina com encarregada de gestionar-ne el retorn.
- 3) Iniciar una crida asíncrona al servei web i mostrar un formulari d'espera al client fins que el servei retorni un resultat.
- 4) Capturar l'event de retorn del servei i analitzar si s'ha cancel·lat el procés o bé s'ha produït un error durant l'execució del servei en l'extrem servidor.
- 5) Tractar la informació rebuda i assignar-la a les entitats corresponents, actualitzar la interfície gràfica,...

Tot seguit, un exemple on podem veure els diferents punts anotats:

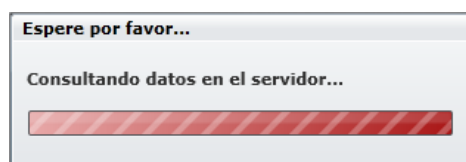
<pre> Private WithEvents M As New MaidenGate.MaidenClient(MainPage._BIND_MAIDEN, MainPage._URL_MAIDEN)  Private Sub ConsultarArticlesNous()     MainPage.setWaitingFormInfo("Consultando artículos pendientes de generar...")     M.ConsultarArticlesNousTemporalsAsync(MainPage._CLI) End Sub  Private Sub endConsultarArticlesNous(sender As Object,     e As MaidenGate.ConsultarArticlesNousTemporalsCompletedEventArgs)     Handles M.ConsultarArticlesNousTemporalsCompleted      If e.Cancelled Then         MainPage.endWaitingForm()         MainPage.ShowCancel()     Else         If Not IsNothing(e.Error) Then             MainPage.endWaitingForm()             MainPage.ShowError(e.Error)         Else             _ARTS = e.Result             _PCV_ARTS = New Windows.Data.PagedCollectionView(_ARTS)             _PCV_ARTS.Filter = New Predicate(Of Object)(AddressOf predFiltrarNoGenerats)             _PCV_ARTS_GENERATS = New Windows.Data.PagedCollectionView(_ARTS)             _PCV_ARTS_GENERATS.Filter = New Predicate(Of Object)(AddressOf predFiltrarGenerats)             CarregarDades()             MainPage.endWaitingForm()         End If     End If End Sub </pre>	<div>1+2</div> <div>3</div> <div>2</div> <div>4</div> <div>5</div>
--	--

És molt important tenir en compte que les crides al servei sempre són asíncrones<sup>42</sup>, ja que si en un moment donat es volen fer més d'una crida consecutiva, caldrà tenir en compte les seqüències d'execució i controlar-les adequadament perquè no perdem la coherència amb el procés complet de crides.

També és important mostrar una pantalla d'espera al client en totes les crides per evitar que es modifiqui l'escenari de la interfície o alguna de les dades que conté mentre s'executa el servei, i que podria provocar alguna incoherència al tractar els resultats.

Per tant utilitzarem unes finestres que prèviament hem prefabricat per poder utilitzar en tota l'aplicació per:

- Mostrar l'espera al client mentre consumim un servei (*winWait*):

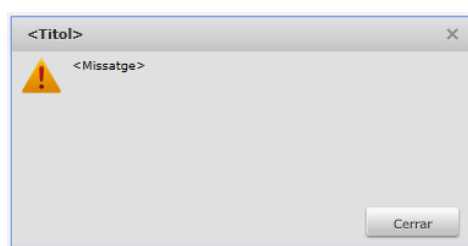


<sup>42</sup> Silverlight Best Practices: Asynchronous Programming – [InformIT](#)

- Avisar sobre un procés cancel·lat o d'un possible error, que en ambdós casos utilitzaran una finestra dissenyada genèricament per poder tractar diferents tipus de missatgeria a l'usuari (*winInfo*):

```
Public Shared Sub ShowCancel()
    Dim Finestra As New winInfo()
    Finestra.SetInfo("Cancelado", "Se ha cancelado el proceso.", winInfo.TipusMissatge.Avis)
    Finestra.Show()
End Sub
Public Shared Sub ShowError(Ex As Exception)

    If TypeOf (Ex) Is System.ServiceModel.FaultException(Of MaidenGate.MercuryErrorControllerErrorSistema) Then
        MainPage.ShowError(CType(Ex, System.ServiceModel.FaultException(Of
            MaidenGate.MercuryErrorControllerErrorSistema)).Detail.Missatge)
    Else
        MainPage.ShowError(Ex.Message)
    End If
End Sub
```



### Estructura de la pàgina principal i validacions inicials

Després de l'enllaç dels serveis amb l'aplicació SetelSys, cal definir el marc de treball de l'aplicació i adaptar-la a la necessitat de poder donar accés a diferents usuaris amb uns permisos concrets d'accés a aplicacions des dels dispositius acreditats.

Per això s'ha desenvolupat una finestra dedicada a l'inici de sessió de l'usuari (*Login.xaml*). En aquesta finestra l'usuari selecciona la botiga d'origen de la connexió, així com el seu nom d'usuari i contrassenya. Al fer clic a "Entrar", s'inicia el consum del servei web per l'autenticació en el servidor de dit usuari. Si es produeix un error d'autenticació es notifica a l'usuari:

I en cas de que la validació sigui correcta es carrega la pàgina principal de l'aplicació:



### Finestra d'avisos

Al fer un clic sobre la icona d'avisos ens notifica sobre alertes que es calculen de manera intermitent en el sistema per detectar elements a corregir o revisar.

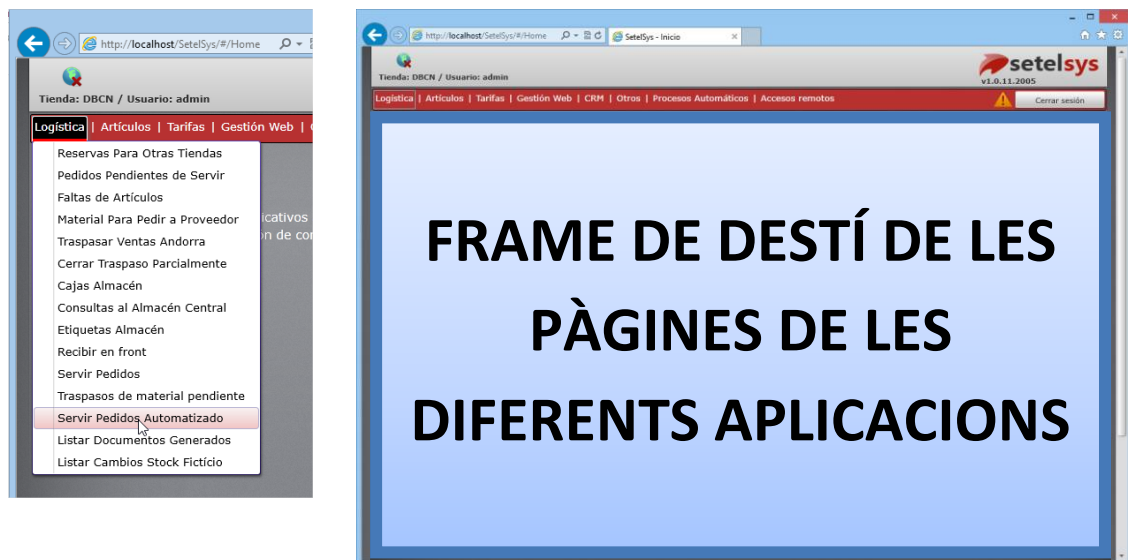




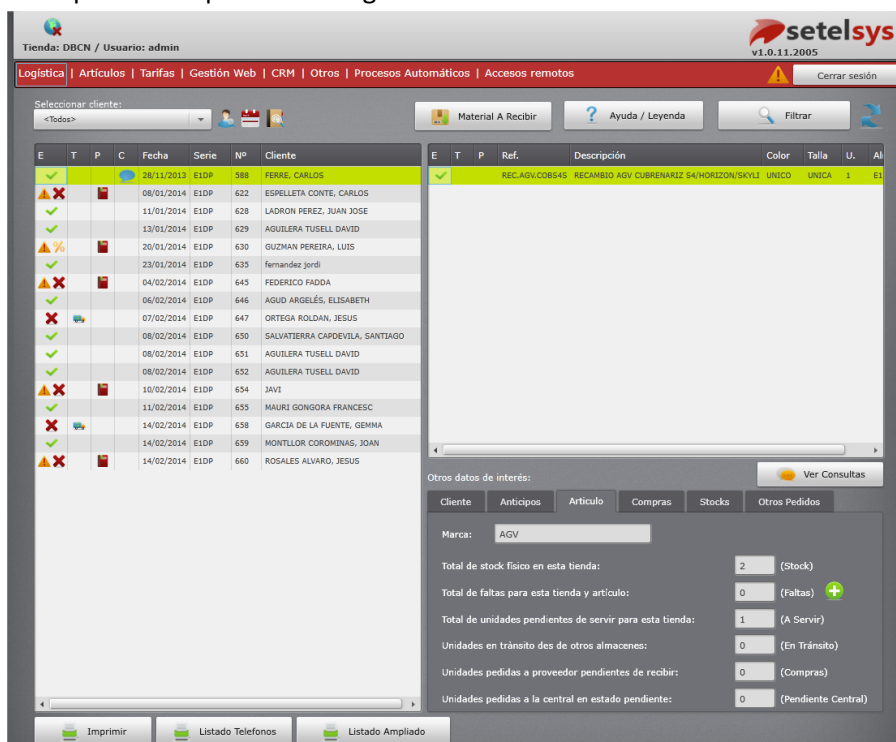
## Navegació per l'aplicació

Una vegada iniciada la sessió i carregats els aplicatius disponibles per l'usuari, aquest pot navegar a través de les diferents aplicacions utilitzant els menús desplegable.

Una vegada seleccionada l'aplicació que es vol executar, aquesta es carregarà sobre el *Frame*<sup>43</sup> destinat especialment per allotjar les diferents pàgines (aplicacions) disponibles:



Exemple d'una aplicació carregada dins del *Frame*:



<sup>43</sup> Navigation Overview in Silverlight – [Microsoft MSDN](#)

## Carrega de les aplicacions autoritzades

Al finalitzar el procés de validació de l'usuari, es consulta a través del servei web, els aplicatius que té autoritzats. En aquest punt es fa us del sistema d'aplicacions i autoritzacions que hem vist en la secció corresponent al **SetelSysRunner**, des d'on podíem assignar permisos a un usuari tant pel que fa a aplicacions, maquines de connexió i botigues.

Pel que fa a les validacions del dispositiu de connexió i la botiga d'origen, es validen durant el formulari d'inici de sessió:

```
Private Sub iniLoginForm()  
Private Sub endLoginForm(sender As Object, e As System.EventArgs) Handles Login.Closed
```

En el cas de que l'autenticació s'hagi realitzat correctament es consulten els aplicatius autoritzats:

```
Private Sub GetAuthOps()  
Private Sub endGetAuthOps(sender As Object, e As MaidenGate.GetAuthOpsCompletedEventArgs) Handles  
M.GetAuthOpsCompleted  
Private Sub EmplenarMenuSuperior(L As List(Of MaidenGate.HeimdalSetelSysDataComAplicatiu))
```

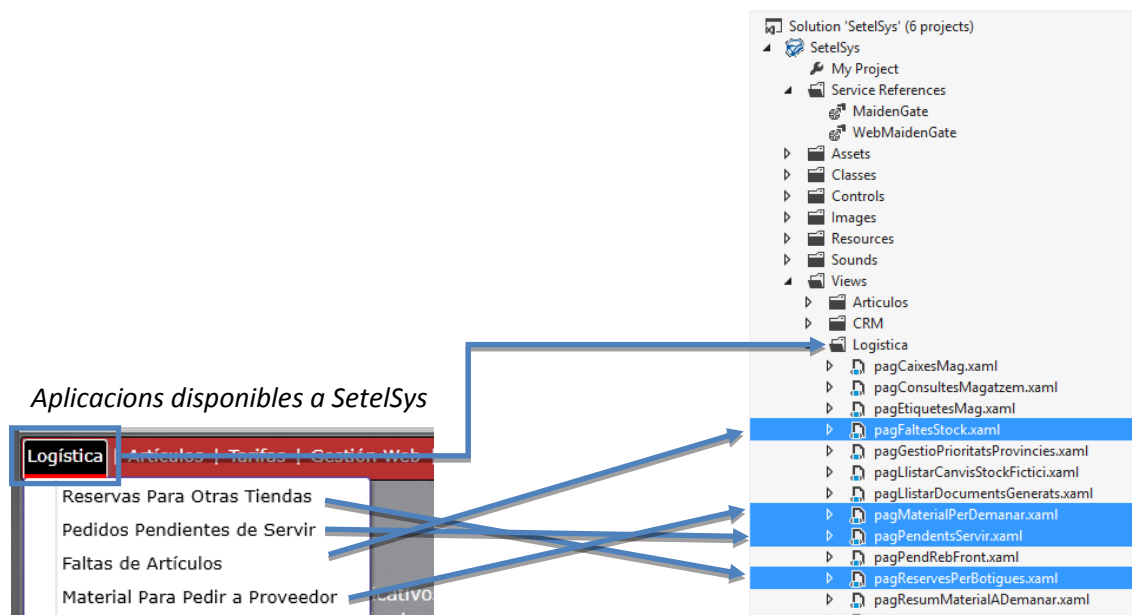
En base dels aplicatius resultants es construeixen els diferents menús d'aplicacions, de forma **100% dinàmica**, que es corresponent amb les famílies d'aplicacions vistes en el **SetelSysRunner**, utilitzant el nom descriptiu també assignat en aquesta mateixa aplicació.

En la següent imatge podem veure el resultat d'una consulta sobre els elements de la base de dades (inserir a través de SetelSysRunner) i la seva relació directa amb les aplicacions disponibles en la interfície del SetelSys:

Valors existents a la taula "aplicatius"			Aplicacions disponibles a SetelSys	
	NomAplicatiu	NomFamilia	NomDescriptiu	
1	ReservesPerBotigues	Logística	Reservas Para Otras Tiendas	Reservas Para Otras Tiendas
2	PendentsServir	Logística	Pedidos Pendientes de Servir	Pedidos Pendientes de Servir
3	FaltesStock	Logística	Faltas de Artículos	Faltas de Artículos
4	MaterialPerDemanar	Logística	Material Para Pedir a Proveedor	Material Para Pedir a Proveedor

De la mateixa manera, s'ha vinculat la funcionalitat de les diferents aplicacions assignades en les diferents pàgines del projecte Silverlight, utilitzant el camp "*NomAplicatiu*" com element comú d'enllaç. Veiem en la següent imatge la relació directa entre el nom de la pàgina implementada i el nom de l'aplicatiu en la base de dades del sistema SetelSys:

## Estructura del projecte i pàgines d'aplicacions implementades



### Agilitat en el desplegament de noves aplicacions

Utilitzant tot aquest protocol aconseguim que implementar una nova aplicació en la interfície del SetelSys, fent-la passar pel sistema de permisos per usuari, sigui tant senzill com:

- Afegir una nova aplicació a través del SetelSysRunner, amb nom **<X>** i descripció **<Y>**, dins la família **<Z>**.
- Donar permisos als usuaris desitjats a dita aplicació **<X>**.
- Implementar una pàgina dins el projecte de Silverlight, dins la carpeta de vistes **<Z>**, amb el nom **"pag<X>.xaml"**.
- Compilem la solució.
- Automàticament ens apareixerà en la interfície del SetelSys (Silverlight), una nova entrada dins del menú **<Z>** amb la descripció **<Y>** que mostrarà la pàgina **"pag<X>.xaml"** al ser seleccionada.

### Carrega i descàrrega de fitxers des del servidor

En certes aplicacions cal enviar fitxers (normalment de text o separats per comes) per executar certs procediments d'automatització o lectura de dades. Per aquest motiu s'ha desenvolupat un parell de finestres especialitzades per la càrrega (*upload*) d'arxius en el servidor, i de descàrrega (*download*) des del servidor.

- *winUpload.xaml* → Càrrega de fitxers
- *winDownload.xaml* → Descàrrega de fitxers

En ambdós casos, el desenvolupament dels fitxers s'ha hagut d'acompanyar de la creació de dos gestors HTTP<sup>44</sup>, un per cada funcionalitat:

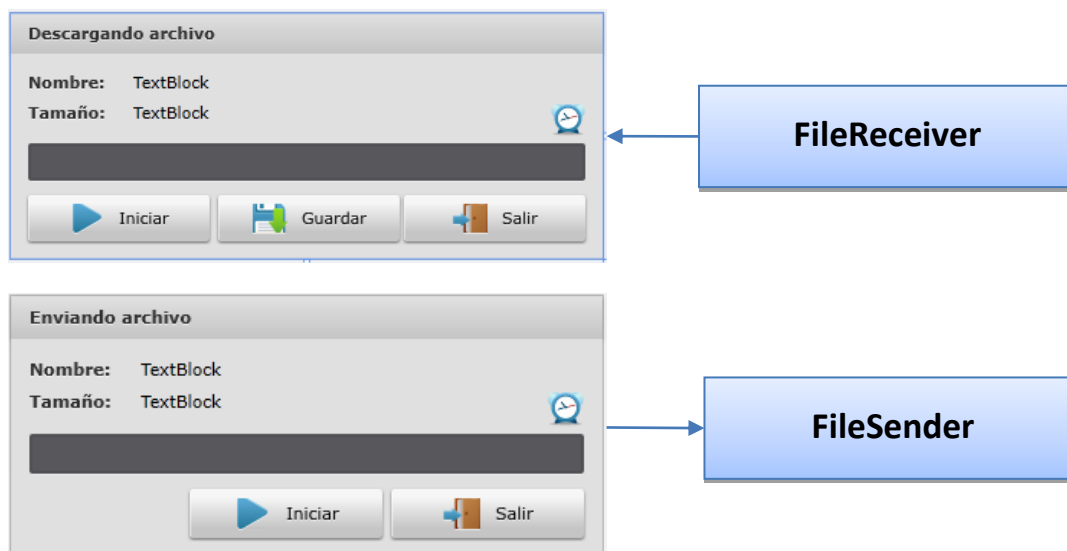
- *FileReceiver.ashx* → Càrrega de fitxers
- *FileSender.ashx* → Descàrrega de fitxers

En podem veure la signatura tot seguit:

```
Public Class FileReceiver Implements System.Web.IHttpHandler
    Sub ProcessRequest(ByVal context As HttpContext) Implements IHttpHandler.ProcessRequest
    ReadOnly Property IsReusable() As Boolean Implements IHttpHandler.IsReusable
End Class
```

```
Public Class FileSender Implements System.Web.IHttpHandler
    Sub ProcessRequest(ByVal context As HttpContext) Implements IHttpHandler.ProcessRequest
    ReadOnly Property IsReusable() As Boolean Implements IHttpHandler.IsReusable
    Private Shared Sub WriteLogFile(Val As String)
End Class
```

Podem veure tot seguit les finestres utilitzades i la relació amb els gestors HTTP utilitzats:



Tant en la càrrega com la descàrrega es permet utilitzar algun dels diferents directoris disponibles en el sistema per diferents utilitats, tots ells situats en l'arrel de l'aplicació allotjada al servidor IIS (*c:\inetpub\wwwroot\SetelSys\tmp*).

Els possibles directoris a utilitzar venen definits per una enumeració:

```
Public Enum DirectorisDescarrega As Integer
    Temporal = 0
    Logs = 1
    RDPs = 2
    PDFs = 3
    SWP = 4
End Enum
```

<sup>44</sup> [HttpHandler in ASP.Net – Code Digest](#)

A l'instanciar alguna d'aquestes finestres, s'especifica ja el flux de dades de l'arxiu local, el nom del fitxer a descarregar, el directori on s'ha d'efectuar el procés, si volem iniciar automàticament la càrrega o descàrrega i si volem que es tanqui de manera automàtica la finestra al finalitzar el procés. Podem veure aquests paràmetres en les seves constructors:

`Public Sub New(ByRef ArxiuLocal As IO.FileInfo, ArxiuRemot As String, Directori As DirectorisCarrega, AutoStart As Boolean, AutoClose As Boolean)`

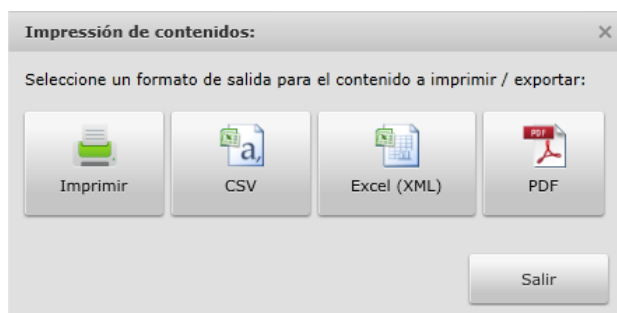
`Public Sub New(ArxiuRemot As String, Directori As DirectorisDescarrega, AutoStart As Boolean, AutoClose As Boolean)`

### Generació d'informes

Silverlight disposa d'unes limitacions en eficiència i funcionalitats importants pel que fa a la creació d'informes. Justament per això s'ha decidit implementar tota la generació d'informes des del servidor, utilitzant els que s'han definit en el nucli **SetelSysCore**, de forma que es generin al servidor i s'enviïn al client o es mostrin directament mitjançant un enllaç web que apunta a una de les carpetes de l'aplicació web del IIS.

Com s'ha vist en l'anàlisi del component **SetelSysCore**, existeixen reports generats amb el propi sistema de Microsoft i també d'altres generats amb Crystal Reports, depenent de les necessitats de l'informe.

Per intentar homogeneïtzar la funcionalitat d'impressió de llistats, s'ha elaborat una finestra exclusivament destinada a comunicar-se a través del servei web per emetre peticions de generacions de llistats. D'aquesta manera en els diferents punts en que es vol imprimir un llistat, simplement cal instanciar una nova finestra de *winConfPrinting*, amb les dades associades i tipus d'informe a generar.



En aquesta finestra ens permet escollir el format de l'arxiu a generar:

- Impressió directa per pantalla.
- Descarregar en format d'arxiu separat per comes (.CSV)
- Descarregar en format Excel (.XLS)
- Descarregar en format d'arxiu portable (.PDF)

Per permetre l'ús d'aquesta finestra per generar diferents tipus d'informes hem optat per sobrecarregar el mètode constructor, de forma que en base del tipus de dades d'entrada i els paràmetres extra, es detecti el tipus d'informe a generar:

SETELSYS	Memòria Projecte Final de Carrera	
----------	-----------------------------------	--

Public Sub New(Dades As List(Of MaidenGate.repLiniaArticlesCreats))

Public Sub New(Pedidos As List(Of MaidenGate.HeimdallERPDataComPedidoPendent), PerRebre As List(Of MaidenGate.HeimdallERPDataComInfoMaterialPerReservar), AutoPrint As Boolean)

Public Sub New(ByRef Pendants As List(Of MaidenGate.HeimdallERPDataComPedidoPendent), IncloureSenseStock As Boolean, BuscarAnticipos As Boolean, BuscarCompres As Boolean)

Public Sub New(Pendants As List(Of MaidenGate.HeimdallERPDataComPedidoPendent), ImprimirConsultes As Boolean)

Public Sub New(ByRef PerDemandar As List(Of MaidenGate.HeimdallERPDataComInfoArticleDemandarProveidor), DataIni As DateTime, DataFi As DateTime)

Public Sub New(ByRef Caixes As List(Of MaidenGate.HeimdallSetelSysDataComCaixa), Tipus As MaidenGate.SeshatTipusReport, Optional Filtre As String = "")

Public Sub New(NomBotiga As String, NumInicial As Integer, NumFinal As Integer, SerieDoc As String, NumDoc As Integer)

Public Sub New(Alarmes As List(Of MaidenGate.HeimdallERPDataComAlarmaMaterial))

Public Sub New(InfoTraspasos As List(Of MaidenGate.HeimdallERPDataComInfoTraspMatPend))

Public Sub New(Pedidos As List(Of MaidenGate.HeimdallERPDataComPedPendentAmbDisponibilitats), IncloureComentaris As Boolean, DescripcionLlistat As String, OnConsultar As MaidenGate.MidgardCentralMTC)

Public Sub New(Canvis As List(Of MaidenGate.HeimdallERPDataComEntradaSP\_FicticiCanviat))

...

## 7.7 Implementació d'eines d'ajuda

Per tal d'ajudar el procés de verificació dels entorns de desenvolupament i producció, així com pel desplegament de noves actualitzacions i versions al sistema en producció, hem elaborat un seguit d'eines per a facilitar-nos en aquestes tasques.

Han suposat una inversió extra de temps, però ens han evitat perdre molt temps en processos repetitius i molt habituals i que per tant a mig termini ens minimitzen el cost temporal de les actuacions.

En concret hem desenvolupat les següents eines d'ajuda:

- **DBComparer:** Compara les definicions de dos bases de dades per tal de poder-ne detectar variacions.
- **DBTester:** Verifica la connectivitat amb les bases de dades del sistema (SetelSys, ERP i Web)
- **FTP Deployer:** Utilitza el servei de FTP com a intermediari pel desplegament de noves versions, copiant tots els arxius necessaris des de la màquina de desenvolupament cap al FTP i posteriorment del FTP en el servidor de producció, situant els arxius descarregats en les rutes especificades en la configuració de l'aplicació FTP Deployer.
- **Chrono Checker:** S'encarrega de monitoritzar l'estat d'un servei. En concret el volem utilitzar per a monitoritzar l'estat de SetelSysWinService, de forma que sempre tinguem un control visual de l'estat del servei.

### 7.7.1 DB Comparer: Comparació de l'estructura de les bases de dades dependents

#### *Objectiu de l'eina:*

- **Analitzar** l'estructura<sup>45</sup> d'una base de dades: Recaptar informació sobre totes les taules i camps de cada una d'elles, així com la tipologia de cada camp (tipus, tamany, claus,...), per posteriorment poder emmagatzemar tota aquesta informació en un arxiu de text.
- **Comparar** dos arxius de text generats a través del punt anterior per detectar totes les diferències existents entre dos bases de dades:
  - Taules existents en una de les bases de dades i no existents en l'altra.
  - Camps existents en la taula d'una de les bases de dades i no en l'altra.
  - Camps diferents en qualsevol dels seus atributs interns a el mateix camp de la mateixa taula de l'altra base de dades.

---

<sup>45</sup> Information Schema Views using Transact-SQL – [Microsoft MSDN](#)

### *Per a què l'hem utilitzat?*

L'ús d'aquesta eina ens ha facilitat controlar tots els canvis de versions en les bases de dades de l'ERP, de forma que podem detectar amb anterioritat possibles canvis en l'estructura de les bases de dades del desenvolupador de l'ERP en els processos d'actualitzacions de versions.

Una vegada detectades dites modificacions podem saber si ens afecten i procedir coherentment per evitar possibles fallades o incoherències en el sistema.

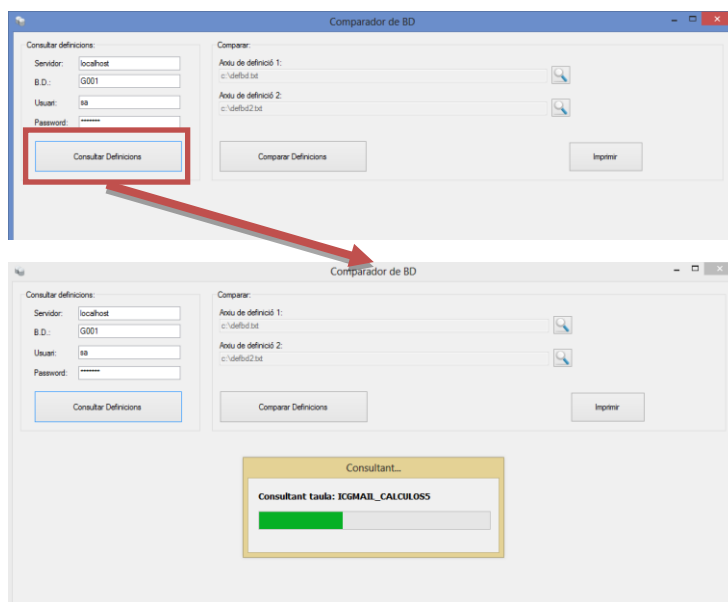
El procediment que hem utilitzat ha estat el següent:

- 1) **Generar** un arxiu de definicions de la BD de l'ERP en el servidor de **producció**.
- 2) **Generar** un arxiu de definicions de la BD de l'ERP en el servidor de **desenvolupament**.
- 3) **Comparar** les definicions de l'entorn de producció amb el desenvolupament actual.
- 4) **Actuar** en conseqüència en el disseny de la solució.

### *Funcionament de l'eina*

A l'iniciar-la ens apareix un formulari on podem escollir les dades del servidor on ens volem connectar (nom del servidor o IP, base de dades i dades de l'usuari acreditat).

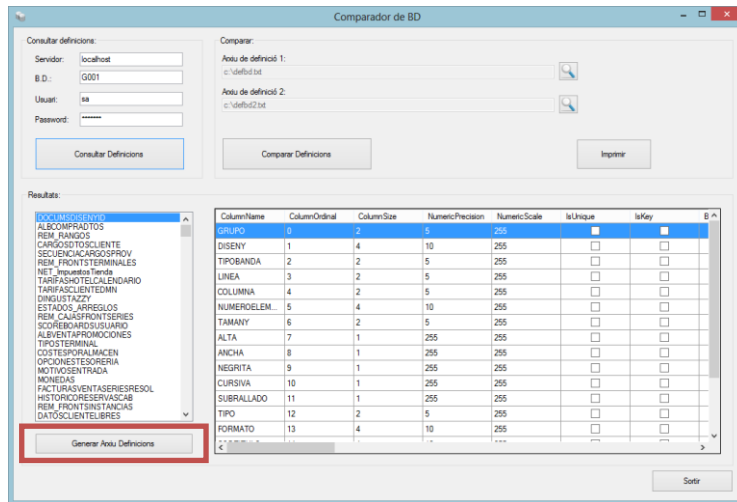
Al prémer sobre "Consultar Definicions", analitza tota la base de dades i ens dona un resultat de les taules existents i la seva definició:



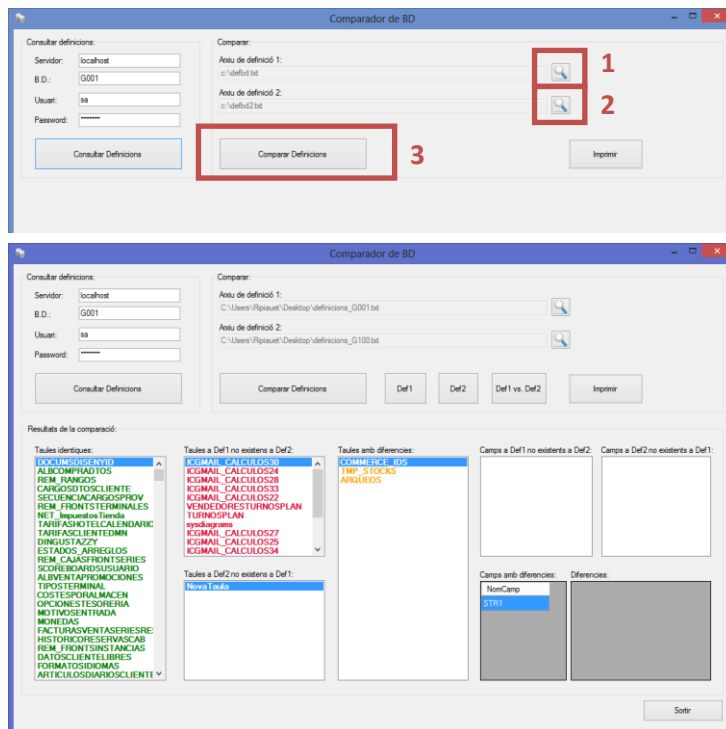
Una vegada finalitzat l'anàlisi de la base de dades ens mostra per pantalla un llistat de taules existents així com la definició de cada una dels camps existents en cada taula.

Podem utilitzar dites dades per analitzar de forma ràpida l'estructura de les taules o també per exportar tota aquesta informació a un arxiu de text:





Per tal de poder comparar dos bases de dades, caldrà especificar primerament els arxius de definició generats amb aquest programa, i posteriorment prémer a “Comparar Definicions”:



El resultat de la comparació entre ambdós arxius de definicions ens permet visualitzar:

- Taules idèntiques
- Taules existents en l'arxiu 1, no existents en l'arxiu 2.
- Taules existents en l'arxiu 2, no existents en l'arxiu 1.
- Taules existents en ambdós arxius, però amb diferències, de les quals podem veure:
  - Camps existents en l'arxiu 1 per dita taula i no existents en l'arxiu 2.
  - Camps existents en l'arxiu 2 per dita taula i no existents en l'arxiu 1.
  - Diferències a nivell intern en la definició dels camps.

### 7.7.2 DB Tester: Verificació de les connexions a la BD

#### Objectiu de l'eina

L'objectiu d'aquesta eina és verificar l'estat de connexió de totes les bases de dades involucrades en el sistema: SetelSys, ERP i Web.

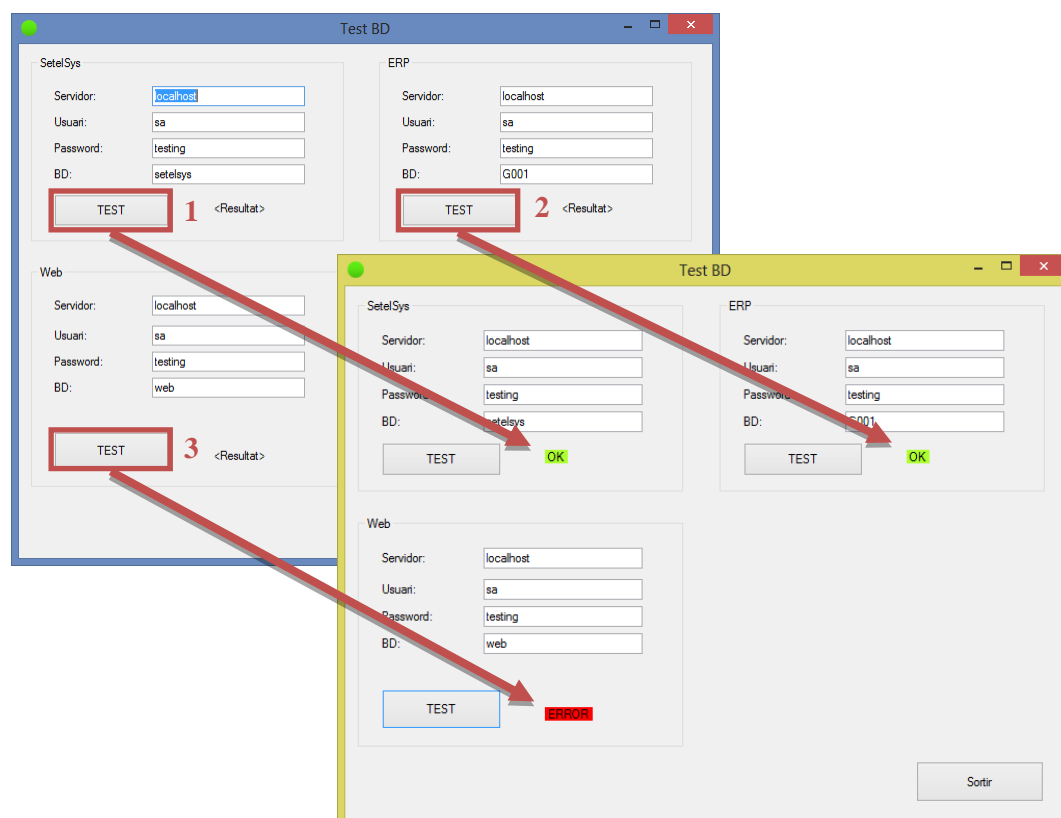
#### Per a que l'hem utilitzat?

L'hem utilitzat per a verificar l'estat actiu i connectivitat del sistema amb les bases de dades quan es detecta algun error de connexió.

#### Funcionament de l'eina

Disposem d'un formulari (amb les dades carregades per defecte de l'entorn de desenvolupament), on podem especificar les dades de cada servidor (fins a 4) dels quals volem verificar la connexió.

Simplement fent clic a tots els botons de test, tenim una imatge immediata sobre la connectivitat amb la base de dades. Aquí podem veure un exemple on la connectivitat amb les bases de dades de l'ERP i de SetelSys és satisfactòria mentre que la base de dades de la web dona error de connexió.



### 7.7.3 FTP Deployer: Per l'actualització del sistema en producció

#### Objectiu de l'eina

Aquesta eina permet comprimir un seguit de fitxers especificats en l'arxiu de configuració de l'aplicació (FTPDeployer.config.xml), utilitzant l'eina 7Zip<sup>46</sup> (requisit de l'eina en qüestió), per a pujar-los en un servidor FTP especificat en l'arxiu de configuració.

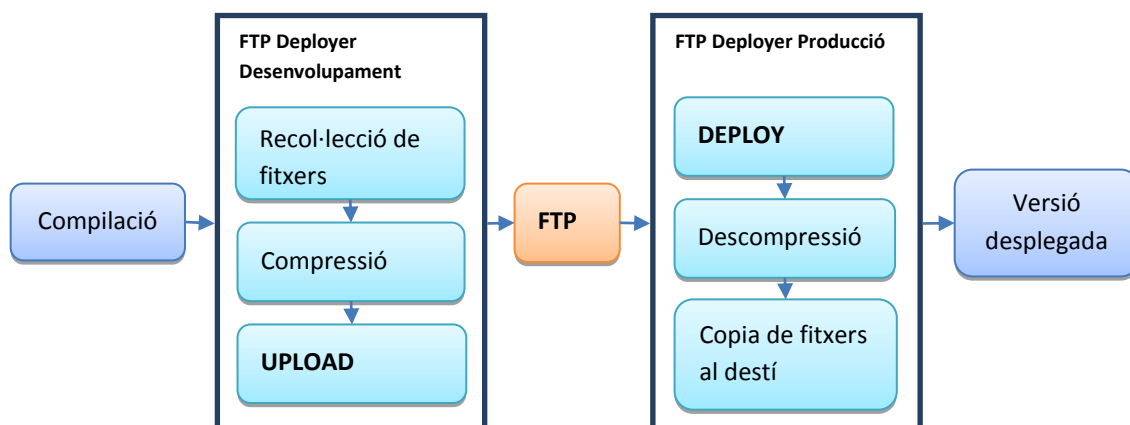
Posteriorment des d'un altre extrem es poden descarregar aquests arxius prèviament pujats, des del servidor FTP especificat, i col·locar-los en el destí especificat en l'arxiu de configuració. Existeix una definició d'origen i destí per a cada combinació de fitxer. Si un dels fitxers configurats per copiar al FTP no es troba s'omet, i si s'ha de descarregar del FTP i no es localitza també s'omet. En cas d'existir en la configuració dels dos extrems, s'utilitzarà l'origen o destí assignat en l'arxiu de configuració de cada extrem.

#### Per a què l'hem utilitzat?

Aquesta eina és una de les més importants i més utilitzades de les que hem esmentat fins ara. S'ha utilitzat per tal de poder desplegar ràpidament noves versions desenvolupades cap a l'entorn en producció. En concret l'utilitzem seguint el següents passos:

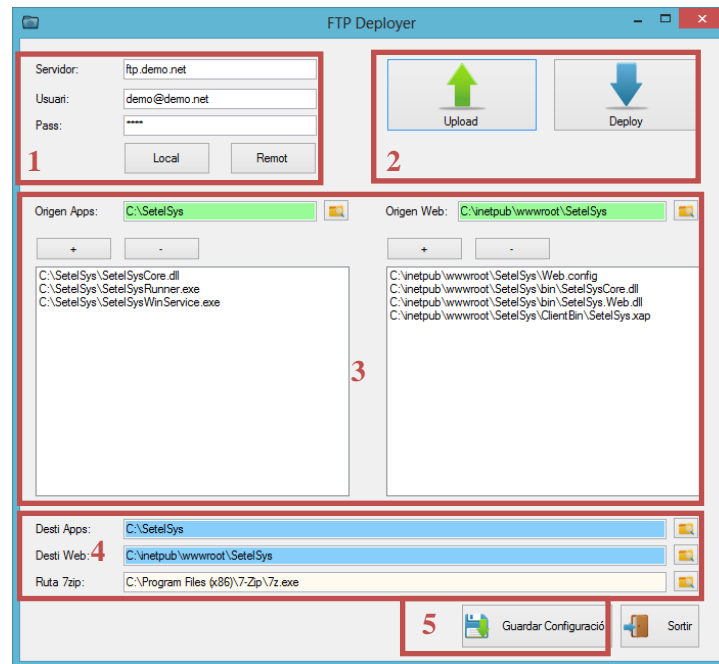
- 1) Compilació de la solució i generació dels objectes i components a actualitzar.
- 2) Existeix en la màquina de desenvolupament una instància del FTP Deployer, configurada per a copiar els arxius prèviament generats cap a un FTP que hem utilitzat com a intermediari (posem dades d'exemple "demo").
- 3) Existeix en el servidor en producció una instància del FTP Deployer, configurada per a descarregar i copiar els arxius existents a l'FTP.
- 4) Executem un UPLOAD des de la màquina de desenvolupament.
- 5) Executem un DEPLOY des de la màquina de desenvolupament.

Aquest procés ens permet, que amb un parell de clics, despleguem la solució actualitzada amb totes les seves dependències al servidor en producció (SetelSysCore, SetelSys, Maiden, SetelSysRunner, SetelSysWinService,...).



<sup>46</sup> Pàgina web del programa 7Zip – [7-zip.ORG](http://7-zip.org)

### Funcionament de l'eina



- 1) Definició dels **paràmetres de connexió** amb el servidor FTP. Es disposa d'un accés directe a les configuracions per defecte a nivell d'ús local (servidor FTP a Internet) o dins la xarxa local en producció (servidor FTP implementat per IIS).
- 2) **Comandes** per la carrega de fitxers a desplegar i per la descàrrega i desplegament de dits fitxers. En desenvolupament farem la càrrega (UPLOAD) i en producció el desplegament (DEPLOY).
- 3) Selecció d'arxius **origen** a copiar, tant pel que fa a la carpeta de l'administració (SetelSysRunner, SetelSysWinService,...), com pel que fa a la carpeta de l'aplicació web Silverlight (SetelSys, SetelSys.Web,...).
- 4) Selecció de les carpetes de **destí** on desplegar la solució i l'accés a l'eina compressora/descompressora 7Zip.
- 5) Per a **guardar** la configuració definida en els punts 1,3 i 4 en l'arxiu de configuració de l'aplicació per no haver de requerir altre cop definir-la a l'iniciar de nou l'aplicació.

#### 7.7.4 Chrono Checker: Verificació de l'estat del servei de Windows en el servidor

##### Objectiu de l'eina

Visualitzar l'estat d'un servei<sup>47</sup> de Windows prèviament seleccionat per tal de saber si aquest es troba instal·lat en el sistema i si està o no en execució, permetent parar o reiniciar el servei. L'aplicació es pot configurar perquè s'iniciï de forma automàtica a l'arrencar l'ordinador.

##### Per a què l'hem utilitzat?

S'ha instal·lat i configurat aquesta aplicació en el servidor per tal de poder monitoritzat l'estat del servei SetelSysWinService, així com poder parar o engegar-lo de forma ràpida i visualment informativa. L'hem configurat perquè s'iniciï amb l'arrencada del sistema operatiu.

##### Funcionament de l'eina

Una vegada iniciat, de forma automàtica es col·loca una icona identificadora en la barra de notificacions del sistema operatiu Windows, de tal manera que si el servei no està iniciat, o no existeix, la icona es mostra en vermell i apareixent / desapareixent de forma intermitent. Si el servei està correctament iniciat es mostra la icona de color verd.

Podem veure-ho exemplificat en la següent imatge:

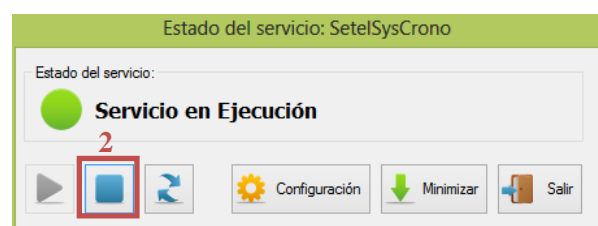


**Servei inexistent o aturat**



**Servei iniciat correctament**

Al fer doble clic sobre la icona de notificació s'obre de forma automàtica la finestra de l'aplicació on ens apareix si el servei està aturat o no existeix al sistema, juntament amb les opcions que anotem a continuació:

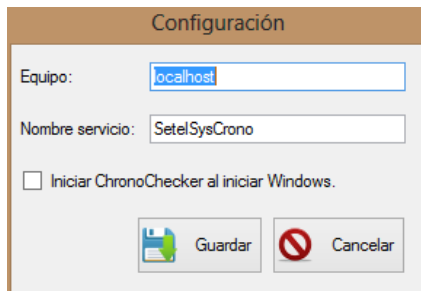


- 1) Iniciar el servei
- 2) Para el servei
- 3) Refrescar l'estat del servei
- 4) Accedir a la configuració de l'aplicació.
- 5) Minimitzar l'aplicació a la barra de notificacions (per tornar-la a activar simplement cal fer-hi doble clic).

<sup>47</sup> Start or Stop a Windows Service Using VB.NET - [DotNetHeaven](http://www.dotnetheaven.com/Start-or-Stop-a-Windows-Service-Using-VB.NET/)

Podem accedir a la configuració de l'aplicació a través del botó que apuntàvem anteriorment, des del qual podrem especificar:

- Equip a monitoritzar (actualment només funciona a nivell d'equip local).
- Nom del servei a monitoritzar
- Activar / desactivar l'arrancada automàtica amb el sistema operatiu.



### 7.7.5 Automatització de la instal·lació del servei de Windows

Per a poder facilitar la instal·lació i desinstal·lació del servei SetelSysWinService, s'han preparat dos arxius de comandes en lot (.BAT) per que s'executin durant els events associats a la pre-compilació i post-compilació del servei, tenint-lo així sempre actualitzat també en l'entorn de desenvolupament:

#### Ordres executades en la pre-compilació:

```
cd c:\windows\Microsoft.NET\Framework\v4.0.30319
if exist c:\SetelSys\SetelSysWinService.exe installutil /u
c:\SetelSys\SetelSysWinService.exe
```

#### Ordres executades en la post-compilació:

```
cd c:\windows\Microsoft.NET\Framework\v4.0.30319
installutil c:\SetelSys\SetelSysWinService.exe
/LogToConsole=true /LogFile=c:\SetelSys\SetelSysWinService.log
```

### 7.7.6 Automatització de còpies de seguretat de tot el projecte

Per tal d'automatitzar còpies ràpides de tot el codi associat a la solució, s'ha programat un arxiu de comandes en lot per a fer-ho.

El que es fa es generar un arxiu comprimit amb 7Zip de tot el contingut de les carpetes de codi, aplicacions i IIS, un per cada agrupació. Els segella amb el dia i hora de la còpia i els mou a la unitat Z que és una unitat de xarxa vinculada a un servidor NAS existent a la xarxa de treball.

El contingut és el següent:

```
@echo off
cd c:\Projectes\SetelSys\
For /f "tokens=1-4 delims=/ " %a in ("%DATE%") do (set mydate=%a-%b-%c)
For /f "tokens=1-2 delims=/: " %a in ("%TIME%") do (set mytime=%a-%b)
"c:\Program Files (x86)\7-Zip\7z.exe" a -mx5 -x!_resources "SetelSys Code Backup %mydate%
%mytime%.7z" c:\Projectes\SetelSys\*
"c:\Program Files (x86)\7-Zip\7z.exe" a -mx5 "SetelSys Apps Backup %mydate% %mytime%.7z"
c:\SetelSys\*
"c:\Program Files (x86)\7-Zip\7z.exe" a -mx5 "SetelSys www Backup %mydate% %mytime%.7z"
c:\inetpub\wwwroot\*

move "SetelSys Code Backup *" z:\Backups\SetelSys\Codi\
move "SetelSys Apps Backup *" z:\Backups\SetelSys\Apps\
move "SetelSys www Backup *" z:\Backups\SetelSys\www\
```

## 7.8 Llibreries i components externs utilitzats

En aquest apartat llistem les diferents llibreries del marc de treball .NET (.NET Framework v4.0.0) que hem utilitzat pel desenvolupament de la solució, així com les llibreries externes al .NET que s'han hagut d'utilitzar, així com el tipus de llicència i cost de les diferents llibreries o components.

### 7.8.1 Components .NET Framework

En aquest cas només llistarem les classes utilitzades dins l'espai de noms del .NET Framework:

Microsoft.Office.Interop.Excel	System.Web.Entity
Microsoft.ReportViewer.Common	System.Web.Extensions
Microsoft.ReportViewer.WinForms	System.Web.Mobile
Mscorlib	System.Web.Services
System	System.Windows
System.ComponentModel.DataAnnotations	System.Windows.Browser
System.Configuration	System.Windows.Controls
System.Configuration.Install	System.Windows.Controls.Data
System.Core	System.Windows.Controls.Data.Input
System.Data	System.Windows.Data
System.Data.DataSetExtensions	System.Windows.Forms
System.Deployment	System.Windows.Forms.DataVisualization
System.Drawing	System.Windows.Input
System.EnterpriseServices	System.Windows.Input.Toolkit
System.Net	System.Windows.Layout.Toolkit
System.Runtime.Serialization	System.Windows.Navigation
System.Security	System.Windows.Theming.Toolkit
System.ServiceModel	System.Xml
System.ServiceModel.DomainServices.Client	System.Xml.Linq
System.ServiceModel.DomainServices.Client.Web	System.Xml.Serialization
System.ServiceModel.Web.Extensions	
System.ServiceProcess	
System.Web	
System.Web.ApplicationServices	
System.Web.DynamicData	

### 7.8.2 Llibreries externes

A continuació citem les llibreries/components utilitzats de forma addicional per l'elaboració de la solució i les eines addicionals:

Nom de la llibreria / component	Tipus de llicència i cost
<a href="#">SAP CrystalReports for Microsoft Visual Studio</a> <i>CrystalDecisions.CrystalReports.Engine</i> <i>CrystalDecisions.ReportSource</i> <i>CrystalDecisions.Shared</i>	Propietaria (EULA)
<a href="#">Excel JetCell</a> DTG.Excel	Comercial <b>175,00 \$</b>
<a href="#">Pdf Sharp</a> <i>PdfSharp</i>	Open Source
<a href="#">Silverlight Reporting</a> <i>Silverlight.Reporting</i>	Microsoft Public License (Ms-PL)
<a href="#">Silverlight Popup Menu</a> <i>SL4PopupMenu</i>	GNU Library General Public License (LGPL)
<a href="#">ThinkSharp FTP</a> <i>ThinkSharp.Components.Net.Ftp</i>	GNU Library General Public License v2 (LGPL v2)



## 8. Testing

En el projecte SetelSys es va optar per a fer un èmfasi especial en la fase de Testing de Software, així com de la qualitat en general. El Testing dóna una informació vital de la idea de la qualitat que té un software o sistema, i per tant, dels riscos que pot implicar una possible baixa qualitat final. I la idea general, és que en base a aquest coneixement, la direcció dels projectes o empreses poden emprendre les accions necessàries sobre una base solida de coneixement.

### 8.1 Metodologia: Test Management Approach (TMap)

El testing estructurat aporta una sèrie d'avantatges, entre les que en podríem destacar les següents:

- Pot ser usat en qualsevol situació, independentment del client o de la metodologia emprada per a desenvolupar el producte.
- Aporta informació i recomanacions sobre qualsevol risc en referència a la qualitat del sistema sota prova
- Es tendeix a trobar els errors en una fase inicial
- Evita nous defectes
- Els productes resultats de les proves (ex. Casos de prova) són reutilitzables
- Tot el procés en el seu conjunt és comprensible i manejable.

En el nostre projecte des de bon principi vàrem optar per una aposta clara i concisa per aconseguir una alta qualitat del sistema a desenvolupar, fet que ràpidament va situar-nos dins el món del testing estructurat. Com s'ha comentat a sobre, el testing no estructura té alguns avantatges interessants, i de fet, en certs moments s'ha dut a terme un testing més semblant a aquest per avaluar d'una primera passada l'estat d'alguns elements.

Actualment al mercat podem trobar moltes metodologies de testing estructurades, però des de bon principi l'opció per a que ens vam decantar és la de seleccionar **TMap (Test Management Approach)** com a metodologia de proves per al nostre projecte. Com és veurà en el següent punt és una de les metodologies més utilitzades al mercat (per exemple, és la metodologia més usada a Holanda i Alemanya), i gràcies al coneixement tant teòric com d'ús laboral en diversos projectes del que disposem, va semblar la millor opció.

### 8.1.1 Què és TMap?

TMap és la metodologia de proves desenvolupada per l'empresa Sogeti. TMap, o Test Management Approach, és una metodologia estructurada de proves per a sistemes d'informació.

La primera versió de la metodologia va ser definida l'any 1995, descrita per Martin Pol, Ruud Teunissen i Erik van Veenendaal. Cap a finals de 2006 una nova versió va ser publicada amb el nom de **TMap NEXT**, escrita per altres autors (Tim Koomen, Michiel Vroon, Leo van der Aalst i Bart Broekman). TMap va ser creada per la divisió holandesa de Sogeti, que forma part del grup Capgemini, i tot i que originàriament va ser un producte holandès, compta amb una extensa expansió al voltant del món, i s'ha traduït al francès, l'alemany i l'anglès.

TMap es pot resumir en quatre punts essencials que ho defineixen:

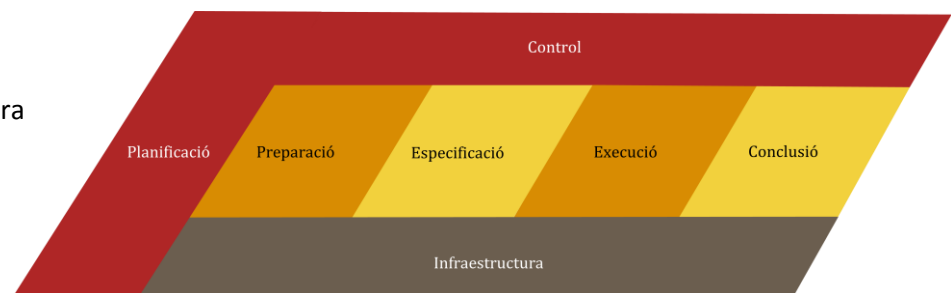


1. TMap està basat en *business driven test management*, o una gestió del test enfocada al negoci.
2. TMap descriu una aproximació estructurada a les proves.
3. TMap conté una caixa d'eines completa.
4. TMap és un mètode de test adaptatiu.

### 8.1.2 Cicle de vida segons TMap

En el cicle de vida de TMap, les activitats de prova es divideixen en set grans fases, i cadascuna de les fases es divideix en una sèrie d'activitats:

- Planificació
- Control
- Infraestructura
- Preparació
- Especificació
- Execució
- Conclusió



Cal tenir en compte que la presentació de les diferents fases en el gràfic, per mitjà d'una paral·lelogram tort no és purament estètic, sinó que ve a representar que en realitat no hi ha cap fase que tingui lloc totalment disjunta de les que l'envolten, sinó que en molts casos les activitats tenen lloc en paral·lel (per exemple, en molts casos l'execució de les proves pot començar en alguna àrea abans d'acabar completament la fase d'especificació).

## 8.2 Aplicació de TMap en el Projecte

TMap, tal i com s'ha explicat ens ofereix el marc metodològic de treball necessari per afrontar la tipologia de projectes que sigui necessari.

En el projecte actual, s'ha optat per a realitzar una breu simplificació ja que per tal i com s'ha treballat conjuntament sent dues persones en l'equip no ens era necessari plantejar-nos realitzar tots els passos.

### 8.2.1 Fase de planificació

Tal i com marca TMap, es va començar amb la fase pròpiament de planificació del Testing. Des del principi, l'abast de les proves a realitzar van quedar molt definides en la fase de proves de Sistema (System Test), proves a nivell funcional principalment, i enfocades a provar les diverses aplicacions que conformen SetelSys (Core, Maiden, Runner, ...) així com les eines d'ajuda (Chrono Checker, BD Comparer, ...).

Es va crear el Pla de Proves per a SetelSys seguint una de les plantilles de TMap.<sup>48</sup>

En aquest document es va anar consolidant tota la informació rellevant a nivell de testing de sistema; fet que el va convertir en un document molt interessant en contenir una bona compilació de informació.

#### *Eina de gestió de proves – Fase de Planificació*

Per a la gestió de proves s'ha optat per l'eina del **Microsoft Test Manager** (MTM), ja que aquesta eina s'adequava perfectament a les necessitats i als principals objectius requerits en el projecte: **Gestió completa de la fase de proves, Traçabilitat, Integració, Automatització.**

### 8.2.2 Fase de preparació

Durant la planificació general del projecte (apartat 4), a l'hora de fer l'Anàlisi de Requeriments ja es va plantejar el realitzar una sèrie de reunions amb el client per tal de clarificar i anar concretant amb més detall com hauria de ser el sistema. Tot i aquesta revisió, durant la fase de preparació es va procedir a una revisió en més detall, on es van detectar una sèrie d'errors menors i incoherències; fet que va permetre corregir-los en una fase molt incipient.

E aquesta fase també es va realitzar la compilació de tota la documentació també es va procedir a la càrrega d'aquests requisits a l'eina de gestió de proves.

---

<sup>48</sup> TMap NEXT Downloads – [Template Test Plan System and Acceptance Tests](#)

### 8.2.3 Fase d'especificació

La fase d'especificació del projecte SetelSys ha estat un dels punts més complexos a treballar, degut a la varietat de components que integra l'aplicació, fet que ha comportat la generació d'una gran quantitat de proves. A mesura que es van anar tancant les especificacions i dissenys tècnics, i un cop realitzada la planificació de la fase de proves, així com al revisió en la fase de Preparació, es va procedir, funcionalitat per funcionalitat ha crear els casos de prova necessaris per a tenir un nivell cobertura de proves correcte per a la qualitat esperada.

Com a punt interessant a mencionar, és el fet que en tot moment s'ha intentat seguir la metodologia TMap en quant a l'especificació de proves, ja que dins de la metodologia podem trobar-hi una bona quantitat de tècniques de disseny de proves que simplifiquen la creació de les mateixes, garantint la cobertura esperada. Algunes de les tècniques utilitzades han estat: **Anàlisi dels valors límits, Proves Sintàctiques, Proves Semàntiques, Testing exploratori, ...**

### 8.2.4 Fase d'execució

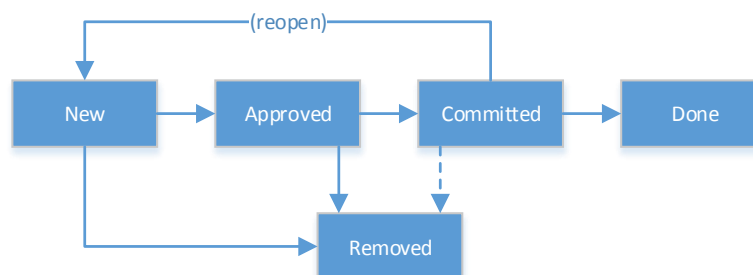
Per a començar les proves ja teníem pràcticament enllestida la fase d'Especificació, amb el que la majoria de casos de prova ja estaven correctament introduïts en el sistema de gestió de proves, el Microsoft Test Manager.

Els resultats de les diferents proves realitzades, tot i anar detectant diversos errors com es veurà més endavant, van ser totalment satisfactòries, i al final de l'execució, i de la validació directament que es va realitzar en l'entorn de pre-producció va permetre afirmar que SetelSys estava preparada per a ser posada en producció, compint amb els [criteris d'entrada i sortida](#) establerts en el Pla de Proves.

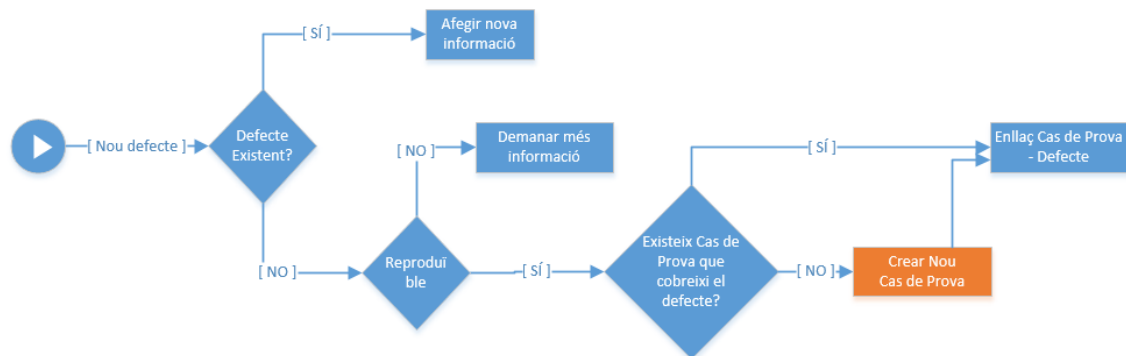
#### 8.2.4.1 Gestió d'errors

Tota la gestió d'errors s'ha realitzat a través de l'eina de gestió de proves de Microsoft, el Test Manager, intentant al màxim aconseguir una correcta traçabilitat, des dels requisits fins als propis defectes passant per les proves.

##### *Cicle de vida dels errors*



*Cicle nous casos de prova en cas d'error.*



### 8.3 Fase de conclusió

En aquesta fase, en la que segons TMap s'ha de **preservar el testware** i avaluar el procés de proves, ja desat correctament tots els jocs de proves presents a l'eina, així com les automatitzacions, per tant de reutilitzar-les de cara al futur.

#### 8.3.1 Fase de Infraestructura

Simplement mencionar en aquest punt que s'ha tingut en compte els entorns propis de proves, i les eines utilitzades han estat: Microsoft Test Manager, Visual Studio / TFS / Coded UI, Test Complete.

#### 8.3.2 Control

Des de bon principi (ja es pot veure fins i tot en la planificació inicial que es va dur a terme), es va creure convenient una gestió molt propera amb el client (fet que TMap també posa de rellevància amb el seu Business Driven Test Management), tant a nivell de reunions per anar controlant el progrés, com per a validar i revisar qualsevol dubte, tant a nivell d'especificacions com per a tenir en compte la qualitat del producte final, i dels possibles errors que podrien restar oberts. A nivell més intern de l'equip, també es tenir en compte la coordinació interna i la necessitat de transmetre la situació de les àrees específiques de cadascú, i per aquest motiu des de bon principi es van convocar reunions cada dues setmanes per analitzar, debatre i plantejar alternatives el progrés del projecte.

## 9. Automatització

### 9.1 Estratègia utilitzada

Seguint dins de la metodologia de TMap, el principal punt que s'ha tingut en compte a l'hora de plantejar les proves automàtiques, en quant a la selecció de proves, és principalment la prioritat de les proves, basades en el risc que tenen per al negoci. A més a més, com a part de l'estratègia de proves automatitzades s'ha valorat molt el fet de minimitzar, de cara el futur, la feina de reescriure les proves; i per aquest motiu, s'ha utilitzat un patró de disseny, tal i com s'explica posteriorment.

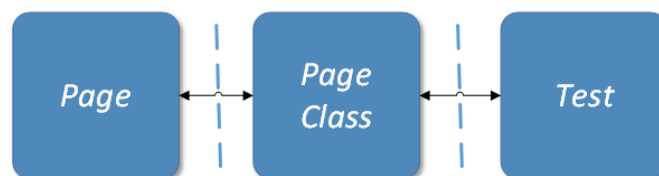
### 9.2 Eines utilitzades

Des del primer moment, vam optar per a treballar de forma integrada amb la suite **de Visual Studio**, utilitzant els **"Coded UI"** per tots aquells tests excepte els de la implementació amb Silverlight. Per altra banda, s'ha optat per utilitzar el **Test Complete**, de SmartBear



### 9.3 Patró de disseny de proves: Page Objects Pattern

El concepte bàsic en què es basa aquest patró és el de representar cadascuna de les pantalles que componen el lloc web o l'aplicació que ens interessa provar, com una sèrie d'objectes que encapsulen les característiques i funcionalitats representades a la pàgina. D'aquesta manera, ens permet consolidar el codi per interactuar amb els elements d'una pàgina en cadascun dels PageObjects.



## 10. Anàlisi econòmic

A continuació es presenta la proposta econòmica desglossada en dues partides: per un costat la part de serveis, que inclou l'elaboració i implantació del projecte, i per l'altre, el hardware necessari per a habilitar l'entorn de producció.

### ***PRESSUPOST SERVEIS***

DESCRIPCIÓ	QTAT	PREU	TOTAL
<b>Desenvolupament del projecte SetelSys</b> Planificació del projecte Anàlisi de Requeriments Especificació i Disseny Tècnic Desenvolupament Testing i Automatització de Proves Finalització i Instal·lació	1487	25,00 €	37.175,00 €
<b>Descompte especial per volum (15%)</b>	1	-5.576,25 €	-5.576,25 €
<b>Obligatorietat de la contractació d'un servei de manteniment durant 24 mesos</b> <b>Cost mensual: 450€ + IVA</b>			
		<b>Total Net</b>	<b>31.598,75 €</b>
		<b>IVA 21%</b>	<b>6.635,74 €</b>
		<b>TOTAL</b>	<b>38.234,49 €</b>

A nivell de serveis hem elaborat una proposta en la que establim una tarifa de 25,00€ + IVA per hora de desenvolupament. En el còmput d'hores hem afegit un 10% de desviació per demores i imprevistos que es puguin produir. Partint de 1352h, afegint aquesta desviació ens queda un total d'hores de 1487h.

Per tal de poder competir amb propostes alternatives de servei de possibles competidors, hem ofert al client, un descompte especial per volum de un 15% sobre l'import base del projecte pel que fa a la part de serveis.

També s'ha vinculat a la proposta, un contracte de manteniment vigent amb un mínim de 24 mesos, amb renovació opcional, per un import de 450,0€ + IVA per mes.

## ***PRESSUPOST HARDWARE***

DESCRIPCIÓ	QTAT	PREU	TOTAL
<b>Server HP Proliant DL360P Gen8</b> Xeon E5-2620 (2.0/12), 8Gb, Controladora SAS HP P420i, DVD	1	1.965,00 €	1.965,00 €
<b>Disc dur hot plug SAS 15.000rpm 500GB Gen8 (6GB)</b>	3	289,00 €	867,00 €
<b>MS-Windows2008 Server Standard. Server + 5CAL</b>	1	535,00 €	535,00 €
<b>Instal·lació i configuració del servidor</b>	1	350,00 €	350,00 €
<b>Opcions</b>			
Mòdul memòria RAM ECC HP 8GB (1 x 8GB)	1	185,00 €	185,00 €
Font d'alimentació redundant HP	1	250,00 €	250,00 €
Extensió de garantia HP CarePack - 3 anys	1	325,25 €	325,25 €
		<b>Total Net</b>	<b>4.477,25 €</b>
		<b>IVA 21%</b>	<b>940,22 €</b>
		<b>TOTAL</b>	<b>5.417,47 €</b>

A nivell de hardware s'ha ofert tots els components i elements opcionals que hem descrit i analitzat en la corresponent [secció del document](#).



## 11. Conclusions

En tota metodologia tant a nivell de desenvolupament com de testing, s'acostuma a finalitzar per una fase de revisió i millora del procés, i no voldríem deixar passar l'oportunitat de realitzar aquesta tasca.

### *Conclusions a nivell de l'aplicació SetelSys:*

Estem molt contents de poder dir que s'ha aconseguit arribar a la satisfacció de la major part dels objectius plantejats en l'inici del projecte. Considerem oportú mencionar-ne els següents:

- Aconseguir unificar un sistema heterogeni, fragmentat i disseminat, plantejant el nou sistema en base d'una arquitectura més coherent i racionalitzada, adaptada a les necessitats reals del dia a dia de l'empresa.
- Les situacions actuals del mercat requereixen d'un desplegament de noves funcionalitats ràpid i eficient, responen a les necessitats canviants de l'empresa. A través de SetelSys s'ha millorat, de forma destacada, el temps en la implantació efectiva d'aquestes noves funcionalitats.
- S'ha assolit l'objectiu de crear un sistema suficientment configurable i senzill en el seu ús com per haver reduït la dependència d'un perfil molt tècnic i especialitzat, dotant al sistema d'una alta interactivitat i parametrització.
- Des de bon principi, l'equip de treball ha tingut clara la importància de mantenir una consciència de qualitat i aplicar amb rigor les normes seleccionades. L'aplicació de metodologies estructurades de proves ha repercutit en un increment de la qualitat final del sistema en termes de: Usabilitat, robustesa, fiabilitat i rendiment.
- Hem dotat al sistema d'un motor capaç de gestionar els errors produïts en l'ús de les diferents aplicacions del sistema, provinents de situacions inesperades que no han estat validades o verificades en els processos de desenvolupament i testing.
- La implementació de les polítiques de seguretat corresponents en l'arquitectura del sistema s'ha traduït en un augment de la seguretat i l'accés controlat.
- S'ha arribat a un nivell de millora del rendiment de la majoria d'aplicacions tant en el temps de còmput com en els resultats gràcies a l'ús d'eines i mecanismes més adequats als processos de negoci de l'empresa i la complexitat algorísmica d'aquests.
- El fet d'haver analitzat en detall tots els processos de l'empresa durant l'anàlisi de requeriments, ha permès a la pròpia empresa detectar i requerir millores en el seu flux de treball.

No obstant hem detectat diferents elements que tot i ser plantejats en una proposta inicial d'objectius, no queden del tot assolits o dibuixen certes mancances:

- La manca del *feedback* adequat amb l'empresa desenvolupadora de la solució ERP i les seves bases de dades de treball, poden causar possibles escenaris d'incoherència en els resultats de certes funcionalitats. Tot i desenvolupar eines per la detecció precoç d'aquestes possibles situacions i establir uns protocols de comunicació interna prèvia instal·lació de qualsevol actualització, no es pot garantir en tot moment la correctesa de la comunicació entre ambdues plataformes.
- La nostra ambició en la creació d'un sistema de control d'errors potser ha estat excessiva, resultant en un sistema sobredimensionat i realment no aprofitat en la gestió d'incidències o detecció d'anomalies.
- Tot i que s'ha mencionat la millora de rendiment global de l'aplicació, per a realitzar un punt d'auto crítica seria interessant mencionar que algunes de les funcionalitats existents requeririen d'un treball de millora i increment del seu rendiment.

### *Conclusions a nivell d'eines de treball*

Ens hem sentit còmodes en la utilització de les diferents eines i solucions de les plataformes de Microsoft, tant a nivell de desenvolupament com de testing, ja que tant per experiències laborals prèvies com per facilitat d'ús, ens ha permès utilitzar-lo amb agilitat.

No obstant i seguint amb el punt d'autocrítica, trobem oportú mencionar que la selecció de Silverlight com a mitjà d'interacció amb l'usuari seria un dels elements que possiblement ens replantejaríem. Tot i que ha aconseguit complir amb les necessitats del projecte i l'entorn, la falta de continuïtat d'aquesta tecnologia pot suposar limitacions en un futur i per altra banda ha implicat l'ús de dues eines d'automatització proves que haurien pogut resumir-se en una.

### *Conclusions a nivell de projecte de final de carrera*

Ja per a finalitzar, no volem passar sense mencionar un parell d'elements que ens han semblat rellevants en quan al projecte de final de carrera:

- Trobem molt interessant el fet de treballar en equip i no a nivell individual ja que aporta un punt de realitat al projecte, que en el dia laboral tots ens trobem.
- El fet de ser un projecte amb un fonament real ens ha permès enfrontar-nos a les dificultats presents en el mercat actual i tant dinàmic. Això ha suposat una sèrie de dificultats afegides que d'altra banda donen un valor molt més alt a la feina realitzada.

### 11.1 Possibles millores futures

Al llarg del projecte i posteriorment amb la implantació, s'han detectat una sèrie de possibles millores tant a nivell d'optimització com d'extensió de la plataforma. Les més rellevants a destacar són les següents:

- Tot i no estar en el full de ruta del projecte, ni en les necessitats actuals del client, s'ha pogut observar els possibles avantatges que suposaria la extensió del sistema en plataformes mòbils. Inclús l'especialització de certes funcionalitats dirigides a l'ús massiu en les botigues i magatzems del client.
- Caldria replantejar alguns elements arquitectònics que dotessin al sistema d'un major rendiment en la carrega inicial de la interfície, per a donar resposta a les mancances pel que fa a connectivitat existents a l'empresa. Al no ser un factor sota el nostre control, caldria adaptar-s'hi.
- Optimitzar el grau d'automatització en el desplegament de noves versions en l'entorn de producció.
- Caldria millorar el control intern en l'execució de processos crítics de llarga duració, per evitar pèrdues innecessàries de temps al re-executar aquest passos i minimitzar l'impacte en els fluxos de treball.
- En certs processos s'ha detectat reiteradament necessitats de parametrització que caldria cobrir per reduir la dependència d'una actuació tècnica.

## 12. Bibliografia i referències

### *Planificació i gestió del projecte*

- [Microsoft Project](#)
- [Toggl](#)

### *Desenvolupament*

- **ASP.NET**
  - Microsoft – [The official Microsoft ASP.NET site](#)
  - ASP.NET – [ASP .NET Web Forms](#)
  - Code Digest – [HttpHandler in ASP.Net](#)
  - MSDN – [ASP .NET Web Configuration Guidelines](#)
- **MVC**
  - Microsoft – [ASP MVC Overview](#)
  - Wikipedia – [Model View Controller](#)
- **Silverlight**
  - Microsoft – [Que és Silverlight?](#)
  - Wikipedia – [Format d'arxius .XAP](#)
  - MSDN – [What is XAML?](#)
  - Tech Republic – [Create rich interfaces with Microsoft Silverlight](#)
  - MSDN – [Create a new Silverlight Project](#)
  - Code Project – [Using WCF Service with Silverlight](#)
  - InformIT – [Silverlight Best Practices: Asynchronous Programming](#)
  - MSDN – [Navigation Overview in Silverlight](#)
  - MSDN – [Silverlight Controls](#)
  - Microsoft – [Microsoft Silverlight 4 Offline Documentation](#)
  - MSDN – [Hosting and consuming WCF Services](#)
- **MVVM**
  - MSDN – [Design Patterns - MVVM - Model View ViewModel Pattern](#)
  - Wikipedia – [Model View ViewModel](#)
- **WPF**
  - Wikipedia – [Windows Presentation Foundation](#)
- **Windows Services**
  - MSDN – [Introduction to Windows Service Applications](#)
  - MSDN – [Create Windows Services](#)

- **Visual Studio i Visual Basic .NET**
  - Microsoft Visual Studio – [Pàgina de presentació de Visual Studio](#)
  - MSDN – [Windows Forms](#)
  - Tutorials Point – [VB.Net – Basic Controls](#)
  - MSDN – [Expression Blend for Visual Studio](#)
  - MSDN – [How Visual Basic Provides Polymorphism](#)
  - MSDN – [Try/Catch Syntax in VB .NET](#)
  - MSDN – [Serialization \(C# and Visual Basic\)](#)
  - MSDN – [Inheritance in VB .NET](#)
  - MSDN – [Parsing XML using XmlReader Class](#)
  - MSDN – [Handling and Raising Events in .NET](#)
  - MSDN – [Creating Windows Form User Control](#)
  - DotNetHeaven – [Start or Stop a Windows Service using VB.NET](#)
  - MSDN – [SqlCommand Class Reference](#)
- **IIS**
  - IIS.Net – [Microsoft IIS \(Internet Information Services\)](#)
- **Microsoft SQL Server**
  - MSDN – [SQL Server Management Studio](#)
  - MSDN – [Microsoft SQL Server 2008 Editions](#)
  - MSDN – [Information Schema Views using Transact-SQL](#)
- **Microsoft Visio**
  - Microsoft Office – [Microsoft Visio 2010](#)
  - Microsoft Office – [Reverse engineer an existing database using Visio](#)
- **Generació d'informes**
  - MSDN – [Microsoft Reports](#)
  - SAP – [Crystal Decisions Crystal Reports](#)
- **Altres**
  - 7Zip.ORG – [Pàgina web oficial de 7Zip](#)
  - MSDN – [RDP \(Remote Desktop Protocol\)](#)
  - Google Sites – [Algorisme de Levenshtein](#)
  - Wikipedia – [Operativa CRUD](#)

#### *Eines de gestió de codi*

- [Apache Subversion](#)
- Tortoise SVN - [Home](#)
- AnkhSVN – [Subversion Support for Visual Studio](#)

## Testing

- **Testing:**
  - Llibre: The Art of Software Testing. Glenford J Myers, 1979
  - Llibre: TMap® NEXT for result-driven testing. Koomen, Tim; van der Aalst, Leo; Broekman, Bart; Vroon, Michiel. 2006
  - The Turing Digital Archive – [Checking a large routine](#)
  - Abstract and Bibliography from NBS Special Publication 500-106, "[Guidance on Software Maintenance](#)"
  - US Department of Transportation - [Developing Functional Requirements](#)
  - Wikipedia – [Language localisation](#)
- **Test Management Approach (TMap)**
  - SOGETI – [TMap Webpage](#)
  - TMap NET - [Downloads](#)
  - TMap NEXT Downloads – [Template Test Plan System and Acceptance Tests](#)
- **Microsoft Test Manager:**
  - [Testing your application using Microsoft Test Manager](#)
  - MSDN Blogs - [Cloning test plans using Microsoft Test Manager](#)
- **Team Foundation Server (TFS):**
  - Visual Studio - [TFS Overview](#)
  - Microsoft Developer Network – [Team Foundation Server](#)
- **Testing de solucions Silverlight:**
  - Microsoft Developer Network - [Testing Silverlight Applications with Coded UI Tests or Action Recordings](#)
- **Automatització:**
  - Selenium Wiki - [PageObjects](#)
  - Microsoft Developer Network - [¿Qué es un patrón de diseño?](#)
- **Automatització amb Coded UI:**
  - MSDN - [Testing a Large Application with Multiple UI Maps.](#)
- **Test Complete:**
  - SmartBear – [Automated Testing](#)
  - Test Complete – [Integration With Visual Studio](#)
  - SmartBeard – [Testing In-Browser Silverlight Applications](#)
  - SmardBeard – [Preparing In-Browser Silverlight Applications](#)